

文章编号: 2095-2163(2022)11-0172-08

中图分类号: TP391

文献标志码: A

基于改进 SOM-K-Means 算法的三维点云分类

邬春学, 胡真豪

(上海理工大学 光电信息与计算机工程学院, 上海 200093)

摘要: 针对目前的点云分类是直接将原始点云作为输入并提前预设点云分类数存在的缺陷, 本文提出一种改进的方法, 在输入前对原始点云进行预处理, 对密集的点云降低密度以减少计算量, 对稀疏的点云进行三角形内部线性插值以便提取完整的特征, 以此提高点云分类的精度。将预处理后的点云数据输入 SOM-K (K-Means 优化的自组织映射神经网络) 模型进行聚类, 再将聚类后的点云数据并行通过 PointNet 网络进行点云数据特征的提取, 这种先进行聚类后、进行特征提取的方法可以充分保留点云在点云空间中的分布特性, 并且不额外增加数据特征提取的计算时间。

关键词: SOM-K-Means 算法; 三维点云分类; 三角形内部线性插值

3D point cloud classification based on improved SOM-K-Means algorithm

WU Chunxue, HU Zhenhao

(School of Optical-Electrical and Computer Engineering, University of Shanghai for Science and Technology, Shanghai 200093, China)

【Abstract】 For the current point cloud classification, the original point cloud is directly used as input and the number of point cloud classifications is preset in advance. This paper proposes an improved method to preprocess the original point cloud before input, and lower the density of the dense point cloud and reduce the amount of calculation, and the sparse point cloud is linearly interpolated inside the triangle in order to extract the complete features, thereby improving the accuracy of point cloud classification. The preprocessed point cloud data is input into SOM-K (K-Means optimized self-organizing map neural network) model for clustering, then the point cloud data after clustering is extracted through the PointNet network in parallel to extract the features of the point cloud data. This method of first clustering and then feature extraction can fully retain the distribution characteristics of the point cloud in the point cloud space, and could not increase the calculation time for special detection and extraction of data.

【Key words】 SOM-K-Means algorithm; 3D point cloud classification; linear interpolation inside a triangle

0 引言

随着现代科技的逐步发展, 人工智能成了各行各业的热点话题与研究对象。在人工智能领域中, 图像分类技术^[1-2]是一种常见的图像处理方式。目前, 基于对二维图像的深入研究, 已有为数可观的科研人员都将研究重点转向了三维图像的分类, 但是由于三维图像包含的信息比二维图像更加复杂多样, 这也导致了三维图像处理过程的时间成本大大增加, 同时也给三维点云数据配准技术带来了不小的挑战。

当下, 有 2 类常见的三维图像分类, 分别是: 三维几何变换分类方法和三维点云分类方法。其中, 三维点云数据具有无序性和稀疏性的等特点^[3]。在传统的点云分类方法^[4-8]中, 是利用点云数据的局部属性来提取人工特征, 但是这样的操作方式使得点云分类的精确度并不高。随着计算机技术的发

展以及深度学习的兴起, 越来越多的研究者尝试将深度神经网络应用于点云数据的分类中^[9-14], 现已取得了令人满意的结果。

在早期对三维点云数据的研究中, 研究人员是将分散在空间中的杂乱的点云数据经过处理转换为用体素格表示, 再通过 3 位卷积对处理后的点云数据进行特征提取。2015 年, Maturana 等人^[15]提出的 VoxNet 方法, 是将深度学习与图像分类相结合, 并且以分布均匀的体素网格作为输入。VoxNet 的思想是用均匀分布的体素网格来代替不规则的点云, 每个网格中包含的点的信息都可以用被占的网格信息来代替。VoxNet 的方法虽然能够解决由于点云自身无序所带来的问题, 但是关于点云自身旋转变化的问题困扰却依然存在。此后, 大量研究者对体素算法^[16-18]进行改进, 却也未能解决计算量庞大的问题。2017 年, Klovov 等人^[19]提出了 Kd-Net 算法, Kd-Net 的思想就是先将整个三维点云数据模型

作者简介: 邬春学(1964-), 男, 博士, 教授, 博士生导师, 主要研究方向: 图像分类、智能家居、物联网技术及应用; 胡真豪(1996-), 男, 硕士研究生, 主要研究方向: 图像分类。

收稿日期: 2022-03-10

哈尔滨工业大学主办 ◆ 专题设计与应用

分层, 然后对每一层的点云数据进行特征提取, 但是采用分层提取的 Kd-Net 算法还是存在一定的局限性, 对于旋转变换后的点云模型以及含有较多离群点和噪点的点云数据模型也未能取得较好的优化结果。2017 年, Qi 等人^[20]也提出了经典的深度学习模型 PointNet, 该模型将点云数据直接作为模型的输入。算法中, 通过对均匀采样后的点云进行训练, 利用点对称函数来减少对于点云自身无序性带来的问题影响, 并通过引入三维旋转矩阵来解决点云自身的旋转问题。仍需指出, PointNet 算法虽然解决了点云自身的问题, 却也仍然存在不足。原始点云经过均匀采样后, 在降低计算复杂度的同时, 却也丢失了点云的部分, 降低了点云分类结果的准确性, 且算法仅以三维点云数据的全局特征作为分类标准参考, 并没有针对三维点云中更加多样化的局部特征进行处理。2018 年, 仍是由 Qi 等人^[21]针对 PointNet 存在的问题, 提出了改进完善的 PointNet++ 网络, 采用分组提取特征的方式, 只是这种方法并没有保留点云空间分布的特征, 且在处理较大规模点云数据时会不可避免地增加计算复杂度。2020 年, 马京晖等人^[22]提出在输入 PointNet 网络前先进行点云预处理和点云聚类, 再并行通过 PointNet 网络, 这样就较好地保留了点云的空间分布特性, 但是研究中使用了 K-Means 聚类, 却会对初始的聚类中心和分类数目比较敏感, 需要提前做出预设, 而且针对不同种类模型预设的分类结果数却都是相同的, 对后续的分类效果影响较大。

综上, 本文在原有的 K-Means-PointNet 网络模型上进行改进优化, 提出了基于改进 SOM-K-Means 算法^[23-27]的三维点云分类算法, 主要研究内容包括:

(1) 对原始点云数据进行预处理, 对于点云密集区域进行简化, 去除冗余的点和离群点, 减少计算复杂度; 对于点云稀疏区域进行稠密化操作, 在稀疏点云中通过插值法进行稠密化操作, 使得稀疏的点云能更好地反映点云模型的特征。针对不同密度的点云采用不同的预处理操作都是为了之后能更好地进行点云分类。

(2) 对于预处理后的点云数据输入到 SOM-K-Means 算法模型, 由 SOM 神经网络先进行粗聚类, 得出输入点云数据的分类数和初始聚类中心, 再用 K-Means 聚类进行精细化聚类。

(3) 为了能够完整保留三维点云数据的特性, 采用并行方式将点云输入 PointNet 网络进行特征提

取。这种并行输入的方式也有利于减少算法的时间。

1 网络结构介绍

1.1 PointNet 结构介绍

PointNet 算法是将原始点云经过均匀采样, 将采样后得到的所有点云数据作为算法的输入, 输入的点云数据第一步操作时与空间变换旋转矩阵 $T-Net$ 相乘来对齐点云数据, 以此来处理点云自身无序的问题。然后将输入的点云数据交给感知机, 在由感知机提取多次特征后, 就将与一个空间变换旋转矩阵 $T-Net$ 相乘来对提取到的点云特征进行第二次的对齐。紧接着, 整合感知机在各个维度上提取到的点云特征, 由最大池化操作来得到点云的全局特征。最后, 将全局特征通过 mlp 来预测分类任务最后的分类数 k , 这里 k 是最后一层的输出数量, 代表分类的个数, 每个类别会对应于点云的分类得分。

对于 PointNet 网络的优点, 文中可做重点阐述如下:

(1) 对原始点云数据进行处理, 保留了原始点云的空间结构特征。

(2) 通过学习到的 $T-Net$ 空间变换矩阵来对齐点云, 解决了点云自身旋转的问题, 并且通过添加损失函数来保证点云的完整性以及对齐后点云的优越性。

(3) 卷积神经网络在提取了每个维度的点云特征后, 通过最大池化层将特征整合成点云的全局特征, 这就有效地解决了点云自身无序的问题。虽然在提取特征时进行了降维操作, 但是在整合后依旧保留了原有的特征。

不仅如此, PointNet 网络除了表现出更多优点外, 同时却也存在着不足: 保留了局部特征、却未做有效利用, 仅以全局特征作为分类参考, 缺少局部特征的参考, 降低了点云数据分类的准确性。

故而, 研究团队就在此基础上进行了二次优化, 提出了 PointNet++ 网络, 解决了忽略局部特征的问题, 但是由于局部特征是 PointNet++ 网络在将点云分层后才提取的特征, 而点云数据分层操作却增加了计算复杂度, 从而增加了算法运行时间。这里给出了 PointNet 算法流程的设计步骤, 顺次表述如下。

输入 一个 $N \times 3$ 的 2D 张量

输出 最后的分类结果

步骤 1 将输入的点云数据通过一个 $T-Net$ 学习到的旋转矩阵来对齐点云。

步骤2 将对齐后的点云经过最大池化操作进行特征提取。

步骤3 使用 *T-Net* 对提取到的特征进行对齐。

步骤4 将对齐后的特征进行最大池化。

步骤5 将全局特征通过最大池化预测最后分类数;将局部特征和全局特征串联,通过最大池化得到每个数据点的分类结果。

1.2 本文算法结构研究

本文提出的算法考虑到 PointNet 网络自身原有的优势并将其保留,保留空间旋转变换矩阵 *T-Net* 来解决点云自身旋转不变性的问题,保留最大池化层将各维度特征整合为全局特征的优势,在此基础上将聚类算法与点云特征相结合,减少运算量,加快运算速度。

首先对原始的点云数据进行预处理操作,将离群点和噪点剔除,将稠密点稀疏化,稀疏点云插值,将预处理后的点云数据输入到 SOM 神经网络中进行聚类,输出分类种数以及粗糙的分类结果。将输出的分类数目作为 k 值,执行 K-Means 算法进行聚类,并对分类后的点云数据进行点云提取,多个类的点云同时进行特征提取,由于此过程为并行运算,并不会额外增加算法的运算量。此后通过最大池化层将提取到的点云特征进行整合,最后将整合的点云特征输入全连接网络,得到分类类别。

改进后的算法克服了原 PointNet 网络忽略局部特征的问题,有效利用了局部特征,大大提高了目标点云分类的准确性。由于局部特征提取的过程为并行计算,所以不额外增加算法运算时间,在运算时间上优于 PointNet++ 网络。本文算法流程如图 1 所示。

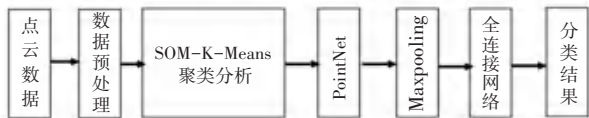


图1 本文算法流程图

Fig. 1 The flow chart of the algorithm in this paper

1.2.1 点云数据预处理

原始的点云数据集中点云数据是杂乱的,是分布不均匀的,因此对点云数据进行预处理操作是有必要的。PointNet 网络模型采用了均匀采样的方式,忽略点云数据的不规则性,有可能降低最后的分类结果准确率。为了能够避免这一可能,本文借鉴数据分析的方法,在预处理阶段对不同密度状态下的点云进行不同的预处理操作,当点云数据在一个

区域内过于密集时,就删减冗余的点云来降低点云密度,缩短运算时间;当局部点云数据过于稀疏时则进行三维点云稠密重建,通过点云插值以保证局部点云形态的完整性,提高分类的精度。本文预处理的设计流程如图 2 所示。

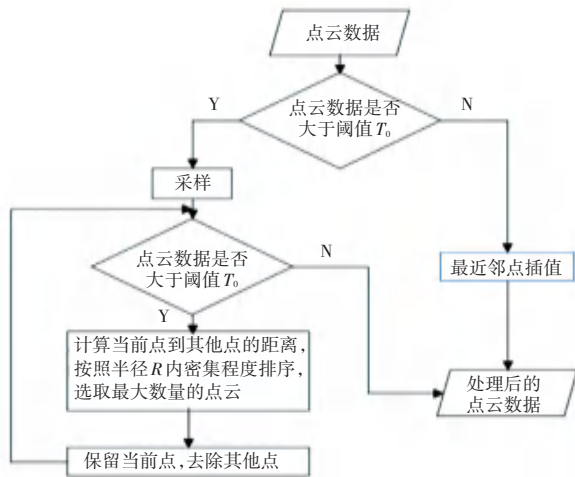


图2 点云筛选流程图

Fig. 2 Flow chart of point cloud screening

1.2.1.1 部分点云高度集中

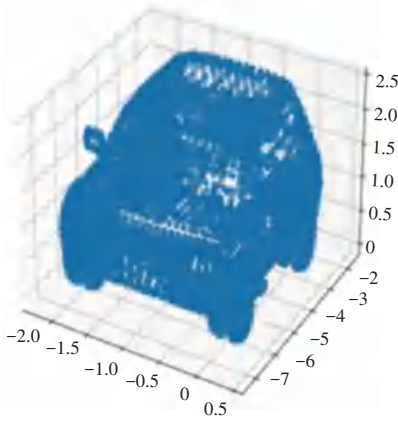
对于点云密集区域,为了将冗余的点云数据去除,采用等间隔的采样方法,计算采样后的点云数据量,判断是否超过目标阈值 T_0 。由于本文改进的算法将与 PointNet 算法进行比较验证,可将目标阈值 T_0 与 PointNet 网络保持一致,设置为 2 048。

当采样后的点云数量大于 T_0 时,计算当前点云数量与 T_0 之间的差值,并且计算当前点云 (x_s, y_s, z_s) 与其他点云之间的距离,计算距离时采用的距离为欧氏距离 D_1 ,对此可表示为:

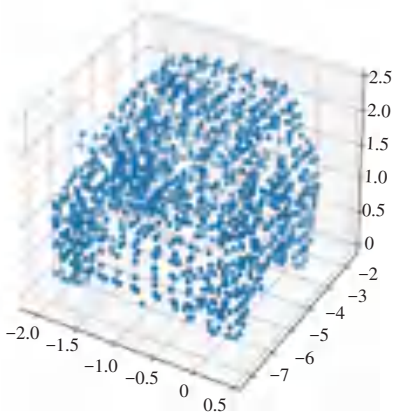
$$D_1 = \sqrt{(x_i - x_s)^2 + (y_i - y_s)^2 + (z_i - z_s)^2} \quad (1)$$

假设点云中的点为圆心, R 为半径划分区域,按照点云区域内点云数量的大小进行排序,将包含最大数量的点云区域与差值进行比较,若不小于差值,则从点云中删除差值大小的点云,保留其他的点云;若小于差值则不对当前点云进行操作,然后重复以上操作,直到点云数量等于阈值 T_0 。

通过此方法可以有效地减少原始点云数据中的冗余点云,实验室效果如图 3 所示。图 3(a)是原始的点云,点云数为 19 686;图 3(b)是经过处理后的点云图,点云数为 2 048。由图 3 可知,此方法可以有效剔除点云集中区域内的冗余点云,且保持点云原有的三维形态。



(a) 原始点云图



(b) 筛选后点云图

图 3 点云筛选对比图

Fig. 3 Point cloud screening comparison chart

1.2.1.2 部分点云区域稀疏

稀疏性是点云自身无法避免的缺陷,如图 4 所示。由于数据集中所包含的点云数量过少,即便可以保持三维点云的外观形态,但是过于稀疏的点云不利于后续的点云分类操作。

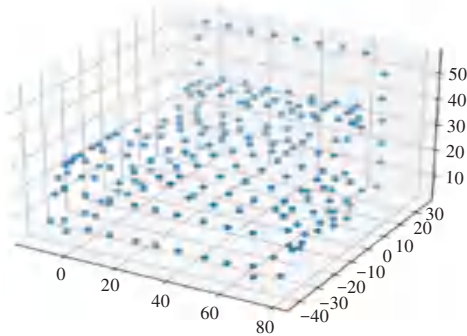


图 4 稀疏点云示意图

Fig. 4 Schematic diagram of sparse point cloud

为了后续点云分类的正确性,需要对稀疏点云进行插值重建,既保持原始三维点云的外观形态,又确保有足够的点云数据用于点云分类操作,本文使

用的插值重建方法是三角形内部插值法,三角形内部线性插值示意如图 5 所示。

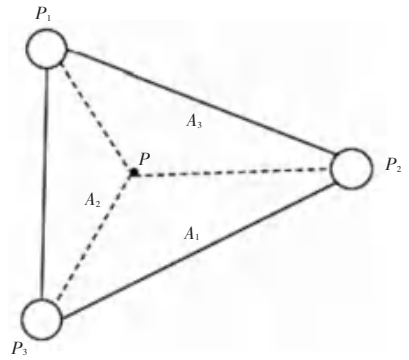


图 5 三角形内部线性插值示意图

Fig. 5 Schematic diagram of internal linear interpolation in the triangle

假设三角形的 3 个定点分别为 P_1, P_2, P_3 , 在三角形内部插入点 P , 并且存在 3 个自由度, 即 $u = \frac{A_1}{A}, v = \frac{A_2}{A}, w = \frac{A_3}{A}$, 使得:

$$P = u \times P_1 + v \times P_2 + w \times P_3 \quad (2)$$

其中, $u + v + w = 1$ 。

稀疏点云重建实验的结果如图 6 所示,对稀疏的点云进行空间插值重建,使点云尽可能地在点云空间中均匀分布。图 6 的实验结果表明,该方法可以有效地克服点云稀疏导致三维点云外轮廓模糊的缺点,促使点云在点云空间中实现均匀分布,有利于后续点云分类计算。

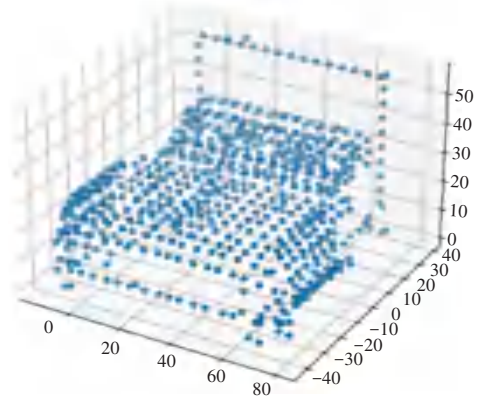


图 6 稀疏点云插值后的结果示意图

Fig. 6 Schematic diagram of results after sparse point cloud interpolation

1.2.2 基于 SOM-K-Means 聚类的三维点云分类

本文在对原始点云进行预处理后,将处理后的点云数据进行 SOM-K-Means 聚类运算,目的是通过聚类算法将点云先做粗分类,同时将点云数据自身的局部特征加以有效合理的利用,同时也避免了

PointNet++网络将点云特征进行分层提取所带来的增加时间复杂度的问题。本文提出的点云分类网

络,如图7所示。

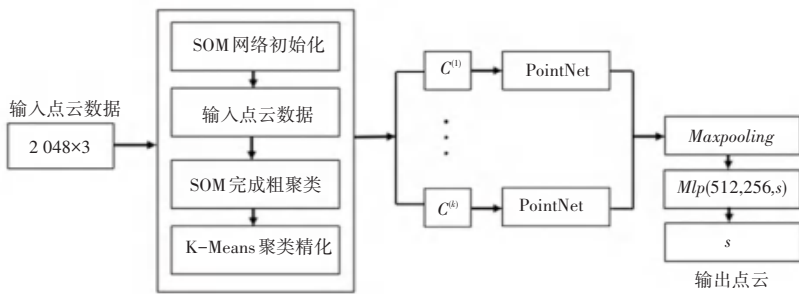


图7 基于SOM-K-Means聚类的三维点云分类

Fig. 7 Classification of 3D point cloud based on SOM-K-Means clustering

图7中,网络输入为原始点云经过预处理后的2048个点,在这2048个点上使用本文设计的SOM-K-Means聚类算法,目的是将这些点划分为 K 个不同的类别,且这 K 个类中每个类别都包含着不等的点云个数,但是这些类别中的点云个数的差距并不大,因为过大的点云数量差距会对后续的计算结果产生较大的影响。然后将这 K 个类的点云数据并行输入到PointNet网络中进行每个类的特征提取。最后利用最大池化层将由这 K 个类中提取到的点云特征进行融合生成全局特征,并将生成的包含局部特征的全局特征输入到感知机中进行分类学习,输出运行的分类结果、即分类类别数 s 。

1.2.2.1 K-Means 算法

在数据分析领域中,K-Means聚类算法是一种常见的算法,该算法可以将数据进行快速有效的粗分类。K-Means算法的本质就是在所有数据点中随机选择 k 个点作为初始的簇中心点,再对其余样本点到这些簇中心点的距离进行计算,将距离同一个簇中心点较近的点都分配到一个簇中,每分配一个样本点,簇中心点都会根据簇中现有的所有点再做一次重新计算。向簇中分配样本点会一直持续到没有剩余的样本点可供分配,此时簇中心点的位置将不会再次发生变化。

该算法的优点是算法简单、收敛速度快。缺点是对初始的聚类中心比较敏感,对于预估合适的 k 值是非常困难的。

1.2.2.2 SOM 算法

自组织映射(SOM)算法是一种高维可视化的无监督学习的聚类算法。SOM神经网络是由输入层和竞争层组成,需要最终聚集的不同类在竞争层中表现为一个个的节点。竞争学习是SOM网络采

用的训练网络的方式,每个输入的样例在竞争层都会与本身相似度最高的节点进行匹配,并将这个匹配度最高的节点称为该输入样例的激活节点。接下来会更新激活节点的参数,并且和激活节点相邻的点也会根据其于激活节点的欧氏距离来对参数进行适当更新。

该算法的优点是,不需要提前预设分类数便可以进行自动聚类,并且容错性高,对异常值和噪声不敏感。缺点是,在训练神经元时会出现个别神经元始终不能胜出、成为节点的情况,导致分类结果的准确性降低,并且SOM神经网络收敛效率较低。

1.2.2.3 SOM-K-Means 模型算法

结合SOM算法和K-Means算法的优缺点,研究者提出了将二者结合的构想并加以实现。由于SOM算法并不需要提前预设分类数,但是在网络收敛时表现不理想;而K-Means算法收敛速度快,但是初始聚类中心和 k 值很难提前预估。因此本文采用SOM-K-Means算法,就是SOM的优化算法,既解决了SOM神经网络收敛效率低、分类结果不准确的问题,又克服了K-Means聚类中心和 k 值的预设问题。SOM-K-Means算法步骤具体如下。

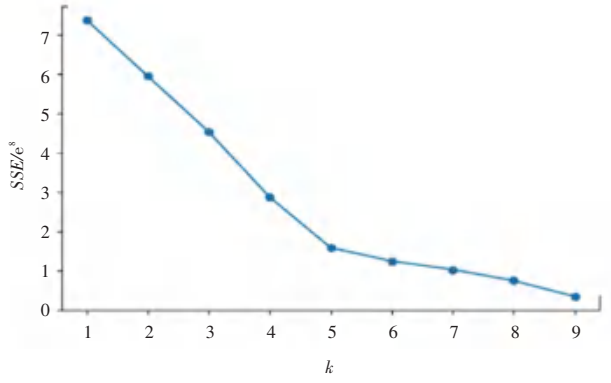
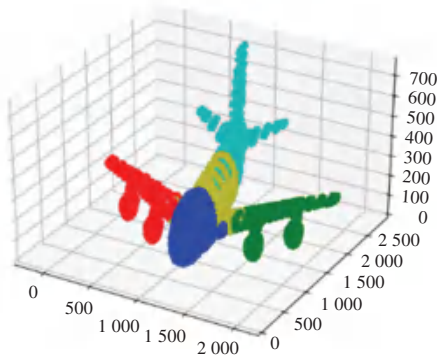
输入 点云数据

输出 聚类结果

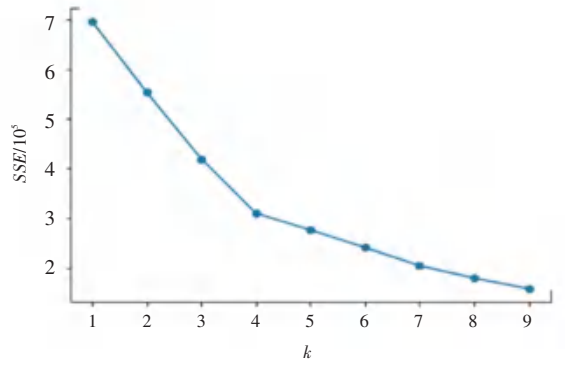
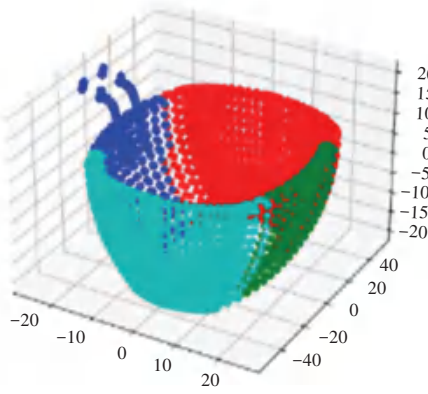
步骤1 将点云数据输入到SOM神经网络中进行聚类,输出分类数目以及初步分类结果。

步骤2 将步骤1中得到的初期分类结果作为 k 值,在预处理过后的点云数据中随机选择 k 个点作为初始的簇中心点,再利用K-Means算法进行聚类。

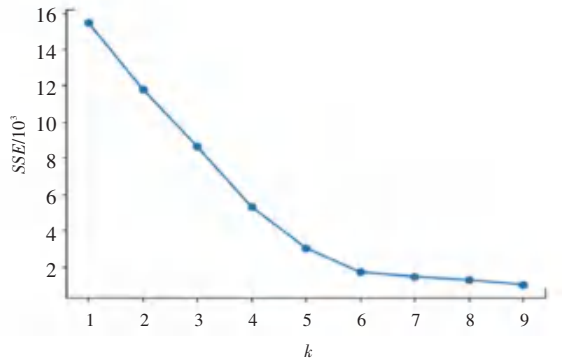
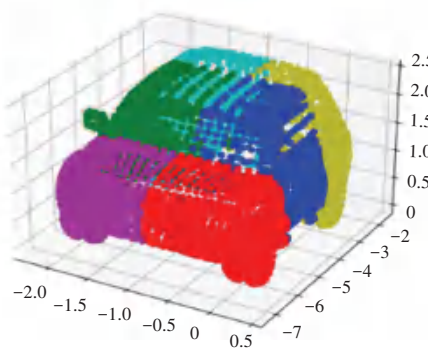
SOM-K-Means聚类算法的结果经过可视化操作后如图8所示。



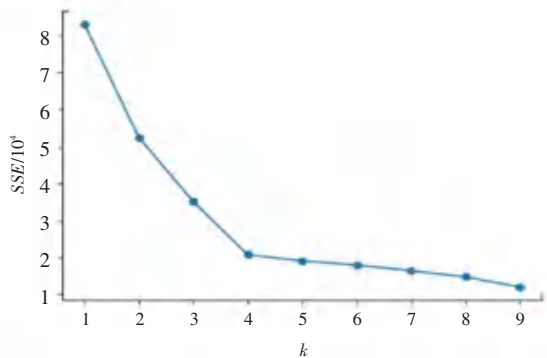
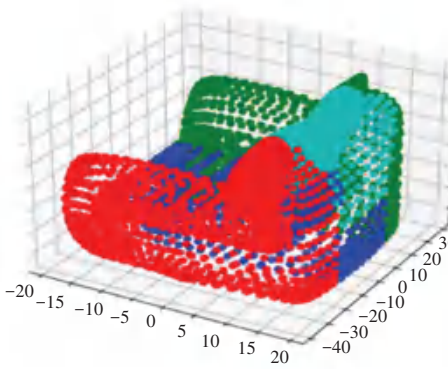
(a) Airplane



(b) Bathtub



(c) Car



(d) Sofa

图 8 聚类可视化结果及其对应 SSE 曲线

Fig. 8 Clustering visualization results and the corresponding SSE curves

1.2.2.4 k 值的验证

手肘法是用来判断 K-Means 算法是否合理的方法之一,计算簇中所有的点到该簇中心点的距离和,即 SSE 值,可以画出 $k-SSE$ 曲线,在折线图中找到所画曲线的拐点。一般来说,曲线拐点就是 K-Means 算法的最佳 k 值, SSE 结果示意图 8。此处需用到的数学公式为:

$$SSE = \sum_{i=1}^k \sum_{x \in C_i} |x - \mu_i|^2 \quad (3)$$

图 8(a)~(d) 中, $k-SSE$ 曲线的横轴为目标聚类数,纵轴为各簇内点到簇中心的距离和。点云数据的分类准确度与聚类数是强相关的,也就是说结果类别越多,点云数据的分类准确度就越高。当预测的 k 值小于实际的最佳分类数时, SSE 的曲线形状相对陡峭;当预测的 k 值大于实际的最佳分类数时, SSE 的曲线形状相对平缓,而在陡峭与平缓之间的拐点对应的 k 值就是该样本数据最佳的分类数。以 Car 模型验证 k 值举例说明,本例中手肘肘部对应的 k 值为 6。对本例数据单独采用 K-Means 算法进行聚类,并画出 $k-SSE$ 曲线,由图 8(c) 可以清晰地看到,当 $k=6$ 时,该点为曲线的拐点,即经过 SOM 神经网络得出的分类数是有实际意义的。

2 实验及结果分析

为了对本文算法的分类效果做出评估,选用和马京晖团队^[22]相同的数据集进行实验,用于训练与测试的数据集来自于公开的数据集 ModelNet10 和 ModelNet40。

2.1 实验条件

实验设备及软件环境见表 1。

表 1 实验配置

Tab. 1 Experimental configuration

操作系统	软件环境	数据集
Windows	Python3.7 TensorFlow-GPU2.4.1	ModelNet10/40

2.2 实验结果对比分析

本文将运用相同的数据集,并在相同的条件下开展实验,对不同算法的分类准确率和算法的运行时间进行记录,再进行对比分析。

2.2.1 准确率比较

本文将模型在相同数据集上的分类准确率与之前的研究者使用的方法进行对比,结果见表 2。从表 2 记录的实验数据中可以看出,本文算法通过原始点数据的预处理,减少了作为输入的点云数量,减

少了算法的运算复杂度,有效降低了计算量。K-Means-PointNet 方法是将 k 设定为统一的固定值进行聚类,再进行特征提取,试验结果表明本文算法在 ModelNet10/40 数据集上对于三维点云分类任务能有效地提高分类准确度,说明对于原始点云的预处理和预处理后的点云数据率先进行聚类,将有助于提升点云分类任务的精度。本文的方法是不固定每个点云数据的分类数,并由 SOM 神经网络分析得出适合输入点云的分类数后再进行 K-means 聚类精化,将聚类后的点云数据并行通过 PointNet 网络,对每个簇中的点云实现提取特征,相比于其他算法,本文提出算法的准确度要更高。将本文算法与 K-Means-PointNet 算法进行对比可以看出,本文提出由 SOM 神经网络计算出合适的分类数比预先设定 k 值对分类精度的细化,能够取得更好的结果。

表 2 分类准确率

Tab. 2 Classification accuracy

方法	输入点云数	准确率	
		ModelNet10	ModelNet40
VotNet	$32^3 \times 1$ (voxel)	92.0	85.9
PointNet	$2\ 048 \times 3$	93.2	89.2
PointNet++	$5\ 000 \times 6$	94.1	91.9
Kd-Net	$2^{15} \times 3$	93.5	91.8
K-Means-PointNet	$2\ 048 \times 3$	94.2	92.6
本文方法	$2\ 048 \times 3$	94.8	93.1

2.2.2 算法用时比较

本文在数据集 ModelNet40 上通过对比 5 种算法训练网络分别需要的时间,来预估每种算法的计算时间成本,结果见表 3。

表 3 训练时间

Tab. 3 Training time

方法	训练时间
PointNet	2~7
PointNet++	23
Kd-Net	120
K-means-PointNet	6
本文方法	5

通过表 3 中的数据可以得出,本文算法在 K-Means-PointNet 的基础上进行改进,但是将点云数据并行通过 PointNet 网络并不影响训练时间,因此本文中提及的网络的训练时间与 PointNet 接近。避免了 PointNet++ 和 Kd-Net 算法多次分层导致计算量庞大、且模型训练时间过长的问题。同时,还避免了 K-Means-PointNet 对 k 值和初始中心的敏感

带来的计算量庞大、训练时间增加的问题,证明本文的改进算法能有效地减少计算量。

3 结束语

本文提出的算法是在 K-Means-PointNet 网络的基础上实现了改进,对原始点云数据进行关键的预处理,而后为了充分利用点云的局部特征,采用 SOM-K-Means 算法将点云进行分类处理,相比于分层提取点云,多个类别的点云并行提取点云特征也不会显著增加其运算时间,所以本文算法在保证运算时间的前提下,有效提高了算法在 ModelNet10/40 数据集上分类任务的准确度,仿真实验运行后得到的准确度分别为 94.8% 和 93.1%。研究可知,在同等条件下,本文算法的准确度要高于其他算法模型,与原有的 K-Means-PointNet 网络相比也有着一定的优化和提升。

参考文献

- [1] 苏赋,吕沁,罗仁泽. 基于深度学习的图像分类研究综述[J]. 电信科学,2019,35(11):58-74.
- [2] 文沛,程英蕾,余旺盛. 基于深度学习的点云分类方法综述[J]. 激光与光电子学进展,2021,58(16):49-75.
- [3] 杨必胜,梁福逊,黄荣刚. 三维激光扫描点云数据处理研究进展、挑战与趋势[J]. 测绘学报,2017,46(10):1509-1516.
- [4] 李红军,刘欣莹,张晓鹏,等. 局部形状特征概率混合的半自动三维点云分类[J]. 浙江大学学报(理学版),2017,44(01):1-9.
- [5] 闫林,刘凯,段玫好. 一种用于点云分类的轻量级深度神经网络[J]. 西安电子科技大学学报(自然科学版),2020,47(02):46-53.
- [6] 程小龙,程效军,郭王,等. 基于激光强度的建筑立面点云分类及信息提取[J]. 同济大学学报(自然科学版),2015,43(09):1432-1437.
- [7] TSENG Y H, WANG Chengkai, CHU H J, et al. Waveform-based point cloud classification in land-cover identification [J]. International Journal of Applied Earth Observation and Geoinformation, 2015, 34: 78-88.
- [8] KIM A M, OLSEN R C, Kruse F A. Methods for LiDAR point cloud classification using local neighborhood statistics [C]//SPIE Defense, Security, and Sensing. Baltimore, Maryland, USA: Society of Photo-Optical Instrumentation Engineers (SPIE), 2013: 873103.
- [9] 宋伯宇,彭玲,池天河,等. 一种改进的全卷积网络激光点云分类方法[J]. 测绘科学,2021,46(05):77-83.
- [10] 胡永东,张正文,李婕. 基于多尺度图卷积的点云分类网络[J].

- 科技创新与应用,2021(08):99-101.
- [11] 王晓红,谌鹏,刘芳,等. 基于集成学习的强鲁棒性三维点云数据分类研究[J]. 包装工程,2021,42(03):252-258.
- [12] 侯向丹,于习欣,刘洪普. 基于图卷积网络的三维点云分类分割模型[J]. 激光与光电子学进展,2020,57(18):204-211.
- [13] 陈盟,陈兴华,邹鹏. 一种面向 3D 点云识别的新型卷积神经网络[J]. 计算机与数字工程,2020,48(05):1196-1200.
- [14] 王本杰,农丽萍,张文辉,等. 基于 Spider 卷积的三维点云分类与分割网络[J]. 计算机应用,2020,40(06):1607-1612.
- [15] MATURANA D, SCHERER S. Voxnet: A 3D convolutional neural network for real-time object recognition [C]//2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). Hamburg, Germany: IEEE, 2015: 922-928.
- [16] WANG D Z, POSNER I. Voting for voting in online point cloud object detection [C]//Proceedings of the Robotics: Science and Systems. Rome, Italy: IEEE Press, 2015, 1: 10-15607
- [17] LI Yangyan, PIRK S, SU Hao, et al. FPNN: field probing neural networks for 3D data [C]//NIPS'16: Proceedings of the 30th International Conference on Neural Information Processing Systems. NY, USA: ACM, 2016: 307-315.
- [18] ENGELCKE M, RAO D, WANG D Z, et al. Vote3deep: Fast object detection in 3D point clouds using efficient convolutional neural networks [C]//2017 IEEE International Conference on Robotics and Automation (ICRA). Singapore: IEEE, 2017: 1355-1361.
- [19] KLOKOV R, LEMPITSKY V. Escape from cells: deep Kd-networks for the recognition of 3D point cloud models [C]//2017 IEEE International Conference on Computer Vision (ICCV). Venice, Italy: IEEE, 2017: 863-872.
- [20] QI C R, SU Hao, MO Kaichun, et al. Pointnet: deep learning on point sets for 3D classification and segmentation [C] //2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Honolulu, HI, USA: IEEE, 2017: 77-85.
- [21] QI C R, YI Li, SU Hao, et al. Point-Net++: deep hierarchical feature learning on point sets in a metric space [C]//NIPS'17: Proceedings of the 31st International Conference on Neural Information Processing Systems. NY, USA: ACM, 2017: 5105-5114.
- [22] 马京晖,潘巍,王茹. 基于 K-means 聚类的三维点云分类[J]. 计算机工程与应用,2020,56(17):181-186.
- [23] 李寒,陶涵斌,崔立昊,等. 基于 SOM-K-means 算法的番茄果实识别与定位方法[J]. 农业机械学报,2021,52(01):23-29.
- [24] 任云,黎洪松. 基于 k-means 聚类的三维 SOM 初始化模式库算法[J]. 桂林电子科技大学学报,2018,38(05):381-384.
- [25] 石丽红. 基于 SOM 算法的高维数据可视化[D]. 秦皇岛:燕山大学,2013.
- [26] 赵文均. 基于 SOM 和 BP 网络的 K 均值聚类算法分析[J]. 电脑知识与技术,2020,16(09):24-26.
- [27] 张石. SOM-K-Means 聚类算法在零售银行客户细分中的应用研究[J]. 攀枝花学院学报,2019,36(05):66-70.