

# 基于轻量级网络和加权 RF 的点云分类分割算法

赵夫群<sup>1</sup>, 郝寒竹<sup>1,2</sup>, 余佳乐<sup>1</sup>

(1. 西安财经大学信息学院, 陕西 西安 710100;  
2. 图木舒克职业技术学院信息工程学院, 新疆 图木舒克 843900)

**摘 要:** 针对点云分类分割方法中存在的高计算开销、复杂网络模型等问题, 提出一种基于轻量级网络和加权随机森林(RF)的点云分类分割算法。该算法采用层次化的方式实现高效分类分割, 首先针对传统神经网络层数多、计算复杂等问题, 构造轻量级神经网络, 并利用其提取点云的全局形状、区域间关系、曲率、法向量和颜色等特征, 实现点云的快速粗分类分割; 然后针对数据不平衡的问题, 设计自适应分类分割策略, 并引入加权 RF, 结合不一致度量筛选与动态加权优化机制, 以实现点云精分类分割。在 ModelNet40 数据集上进行分类实验, 在 Semantic3D 数据集和室外场景点云数据上进行了分割实验, 结果表明, 相比 Local Geo-Transformer, PointNeXt 和 FastPointNet++等算法, 该算法的分类分割精度分别提高了约 1.9%, 1.6%和 1.7%, 分类分割时间分别降低了约 40%, 30%和 20%。由此可见, 基于轻量级网络和加权 RF 的点云分类分割算法在保持较高分类分割精度的同时, 可以有效缩短模型的训练时间, 提高分类分割效率, 是一种有效的点云分类分割算法。

**关 键 词:** 点云分类分割; 轻量级网络; 加权随机森林; 全局形状; 曲率

中图分类号: TP 391.41; TP 18

DOI: 10.11996/JGj.2095-302X.2026010143

文献标识码: A

文章编号: 2095-302X(2026)01-0143-09

## A point cloud classification and segmentation algorithm based on lightweight networks and weighted RF

ZHAO Fuqun<sup>1</sup>, HAO Hanzhu<sup>1,2</sup>, YU Jiale<sup>1</sup>

(1. School of Information, Xi'an University of Finance and Economics, Xi'an Shaanxi 710100, China;

2. School of Information Engineering, Tumushuke Vocational and Technical College, Tumushuke Xinjiang 843900, China)

**Abstract:** To address the issues of high computational cost and complex network models in point cloud classification and segmentation methods, a point cloud classification and segmentation algorithm based on lightweight networks and weighted Random Forest (RF) was proposed. The algorithm achieved efficient classification and segmentation in a hierarchical manner. Firstly, to address the issues of multiple layers and complex computation in traditional neural networks, a lightweight neural network was constructed to extract point cloud features such as global shape, inter-regional relationships, curvature, normal vector, and color, thereby achieving rapid rough classification and segmentation of point clouds. Then, to address data imbalance, an adaptive classification and segmentation strategy was designed. By introducing a weighted RF and combining inconsistency-measurement screening with dynamic-weighting optimization mechanisms, fine classification and segmentation of point clouds were achieved. The algorithm conducted classification experiments on the ModelNet40 dataset and segmentation experiments on the Semantic3D dataset and outdoor-scene point-cloud data. The results showed that compared with Local Geo-Transformer, PointNeXt, and FastPointNet++, classification and segmentation accuracy increased by

收稿日期: 2025-02-21; 定稿日期: 2025-07-23; 通信作者: 赵夫群, E-mail: fuqunzhao@126.com

Received: 21 February, 2025; Finalized: 23 July, 2025; Corresponding author: ZHAO Fuqun, E-mail: fuqunzhao@126.com

基金项目: 国家自然科学基金(62271393); 陕西省教育厅科学研究计划项目(25JS049)

Foundation items: National Natural Science Foundation of China (62271393); Scientific Research Program Funded by Education Department of Shaanxi Provincial Government (25JS049)

approximately 1.9%, 1.6%, and 1.7%, respectively, while classification and segmentation time was reduced by approximately 40%, 30%, and 20%, respectively. Thus, the proposed point-cloud classification and segmentation algorithm based on lightweight networks and weighted RF can effectively reduce the training time of the model and improve the efficiency of classification and segmentation while maintaining high accuracy, making it an effective point cloud classification and segmentation algorithm.

**Keywords:** point cloud classification and segmentation; lightweight network; weighted random forest; global shape; curvature

随着三维传感技术的快速发展,高精度、大场景点云数据的获取日益简单,已在自动驾驶、机器人导航和智能制造等领域<sup>[1-3]</sup>得到了广泛应用。然而,点云数据因其无序性以及高维的特点,为分类分割算法带来了巨大挑战。为了在实际应用中取得良好的性能,点云分类分割算法不仅需要具备高效性和准确性,还必须在资源受限的环境中运行,如小型嵌入式设备对网络的计算复杂度、参数量以及运算速度提出了更高要求<sup>[4]</sup>。本研究旨在满足点云模型的高层语义理解需求,探讨适用于点云模型分类分割方法,并提出一种满足实际应用需求的深度学习网络。

近年来,基于深度学习的点云分类分割方法取得了显著进展。WEN 等<sup>[5]</sup>提出了点云语义分割的注意力网络,该模型利用全局上下文信息和局部细节,有效增强了模型在复杂点云场景中的表现;ZENG 等<sup>[6]</sup>将点云看作图结构,点作为节点,邻近点之间的关系作为边,提出一种基于图卷积网络随机图(Random Graph Based on Graph Convolution Network, RG-GCN),并将其应用到点云分类分割中;LIU 和 TIAN<sup>[7]</sup>将 GateNet 和 PointNet++相结合,提出一种能更准确地提取局部特征的 G-PointNet++算法。但是,基于深度学习的点云分类分割网络往往需要复杂的网络结构,导致计算成本高、训练时间长,因此 LI 等<sup>[8]</sup>集成了多尺度特征融合机制和多层次注意力机制,提出了一种轻量级自适应分类分割方法 CloudDeepLabV3+; JING 等<sup>[9]</sup>仅在局部区域执行注意力机制,提出一种基于 Transformer 的轻量级 Local Geo-Transformer 网络; QIAN 等<sup>[10]</sup>提出的 PointNeXt 通过残差多层感知机(Multi-Layer Perceptron, MLP)和反向最近邻采样实现了精度和效率的平衡;还有 CHEN 等<sup>[11]</sup>结合全局与局部特征并行实现目标分类与分割,并通过统计调整阈值保持恒定误报率。虽然轻量级网络可以减少计算成本,但在处理不平衡数据和复杂场景时表现不足,因此 ALFIO 等<sup>[12]</sup>使用 RF (Random

Forest)算法对城市场景的点云进行分类分割; ZHAO 等<sup>[13]</sup>通过多特征融合和加权随机森林提高了点云分割精度,但计算复杂度较高; MOHAMED 等<sup>[14]</sup>结合 RF 分类器,提出了一种基于机器学习的移动 LiDAR 数据分类分割方法。虽然 RF 算法在点云分类分割中展现出效率优势,但传统的 RF 算法难以有效构建点云数据间的上下文关系,在复杂场景下的适应性较弱。

综上所述,轻量级网络和传统 RF 方法各有优劣,为兼顾计算效率和对不平衡数据的处理能力,本文提出一种基于轻量级网络和加权 RF 的点云分类分割算法。该算法通过轻量级神经网络提取全局和局部特征进行粗分类分割,再采用不一致度量筛选和动态加权机制优化的加权 RF 进行精分类分割,该算法在确保良好分类分割精度的同时,能够有效降低计算资源的占用,展现出良好的实用性和推广潜力。

## 1 基于轻量级神经网络的粗分算法

粗分算法采用轻量级神经网络,通过提取点云的局部几何特征、中层区域特征和全局结构特征实现点云初始分类分割。

### 1.1 网络框架设计

在大规模点云分类分割任务中,传统的神经网络模型由于层数多、计算复杂,难以满足实时性要求。因此,为提高效率和减少计算成本,本文网络框架采用轻量级神经网络进行设计,轻量级神经网络是指结构设计精简、参数数量少和计算量低的神经网络,通常用于在计算资源有限的设备上执行任务,如移动设备、嵌入式系统等。这类网络在设计时会权衡模型的复杂性和精度,以降低内存和计算需求,同时尽量保持较高的任务性能<sup>[15-17]</sup>。轻量级神经网络的框架如图 1 所示。

轻量级神经网络模型的设计策略如下:

1) 简化网络结构。为了保证分类分割性能

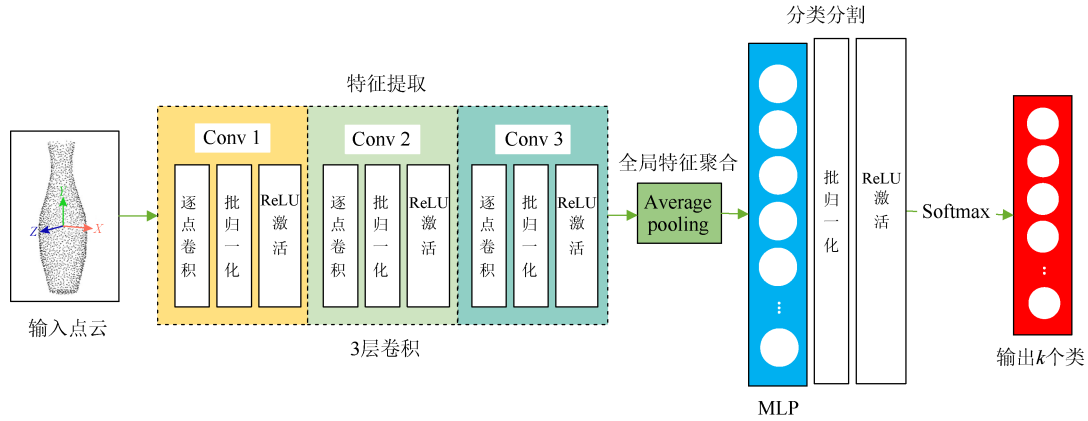


图 1 轻量级点云特征提取网络的框架

Fig. 1 Framework of lightweight point cloud feature extraction network

的前提下降低计算量,可以减少卷积层以及 MLP 的层数。相比复杂的多层网络结构,使用较少层卷积和 MLP 进行特征提取,既能提取足够的点云特征,又能有效控制模型的计算开销。

2) 使用逐点卷积。点云数据的维度通常包括空间坐标(如  $x, y, z$ )和其他特征。在处理特征时,逐点卷积可以针对每个点的特征维度进行计算,无需处理相邻点之间的关系,从而避免了多余的计算复杂度。采用  $1 \times 1$  的卷积核既能保证特征提取效果,又能大幅减少参数数量。

3) 全局池化策略。在完成初步的特征提取后,使用全局平均池化操作将每个点的特征聚合成一个全局特征。这样不仅减少了 MLP 的输入维度,降低了计算复杂度,还保留了每个特征维度的重要信息。因此全局池化策略在保持特征完整性的同时有效减少了参数数量。

4) 合理选择激活函数。激活函数的选择对模型的计算效率有很大影响。选用计算成本低的 ReLU 激活函数,可避免因复杂激活函数带来的高计算开销。使用 ReLU 可以在每层中减少激活操作的复杂性,从而保持模型轻量化,确保其高效运算。

## 1.2 特征提取

在轻量化网络设计中,每一层卷积都会从输入数据中提取不同层次的特征。卷积层通过卷积核在局部区域内扫描输入数据,从中提取空间特征。卷积核大小为  $1 \times 1$  (用于逐点卷积),适合点云分类分割,可逐个点提取局部特征。

假设输入的数据为  $\mathbf{X} \in \mathbb{R}^{C_{in} \times N}$ ,其中  $C_{in}$  为输入通道数,  $N$  为点的数量,卷积核  $\mathbf{F} \in \mathbb{R}^{C_{out} \times C_{in} \times 1}$ ,其中  $C_{out}$  为输出通道数,进行逐点卷积,卷积的计算式为

$$\mathbf{Y}[:, j] = \sum_{i=1}^{C_{in}} \mathbf{W}[:, i] \cdot \mathbf{X}[i, j] + b \quad (1)$$

式中:  $\mathbf{Y} \in \mathbb{R}^{C_{out} \times N}$  表示输出特征图;  $\mathbf{Y}[:, j]$  表示第  $j$  个点的输出特征向量;  $\mathbf{W} \in \mathbb{R}^{C_{out} \times C_{in}}$  表示卷积核权重;  $\mathbf{W}[:, i]$  表示对应于第  $i$  个输入通道的权重向量;  $\mathbf{X}[i, j]$  表示第  $i$  个通道、第  $j$  个点的特征值;  $b \in \mathbb{R}^{C_{out}}$  表示每个输出通道的偏置值。

在此采用 3 层卷积结构,其中第一层捕捉点云的局部几何结构,如点之间的距离关系或位置模式。计算式为

$$\mathbf{Y}^{(l)} = \text{ReLU}(\mathbf{X}' * \mathbf{W}^{(l)} + \mathbf{b}^{(l)}) \quad (2)$$

式中:  $\mathbf{Y}^{(l)}$  表示第一层的输出,  $\mathbf{Y}^{(l)} \in \mathbb{R}^{C_{out} \times N}$  将作为下一层的输入; ReLU 激活为输出引入非线性,  $\text{ReLU}(x) = \max(0, x)$ , 输入  $\mathbf{X}'$  和第一层权重  $\mathbf{W}^{(l)}$  计算线性变换;  $*$  表示卷积操作;  $\mathbf{b}^{(l)}$  表示第一层偏置项。

在局部特征获取后,卷积第二层会捕捉更多全局结构信息,如局部形状的组合(例如平面或曲面)。第二层的输出为特征图,并保留了每个点的局部信息。最后第三层卷积会生成高维度的表示,描述整个点云的全局特征。这些特征在最终将通过全局平均池化后进行汇总。全局平均池化计算式为

$$\mathbf{Z} = \frac{1}{N} \sum_{i=1}^N \mathbf{Y}_i^{(l)} \quad (3)$$

式中:对所有点的特征求平均,以获得整个点云的全局表示,  $\mathbf{Z} \in \mathbb{R}^{C_{out}}$  表示全局特征。

在全局平均池化后,再使用第一层 MLP 将卷积层提取的全局特征映射为紧凑的特征表示。类似的,计算式为式(2),但此时的  $\mathbf{X}''$  表示 MLP 的输出特征。之后,再将特征映射为  $k$  中粗分结果。映射的计算式为

$$\hat{Y} = \text{Softmax}(X'W_{\text{cls}} + b_{\text{cls}}) \quad (4)$$

$$\text{Softmax}(x_i) = \frac{e^{x_i}}{\sum_j e^{x_j}} \quad (5)$$

式中： $W_{\text{cls}} \in \mathbb{R}^{256 \times k}$  表示分类分割层的权重； $b_{\text{cls}}$  表示分类分割层的偏置； $\text{Softmax}$  表示归一化函数，用于将特征映射为概率分布。

综上所述，在经过特征提取后，将特征映射为分类分割结果，从而实现点云的初始粗分类分割。再采用加权 RF 算法，对点云进行进一步的精分类分割。

## 2 基于加权 RF 的点云精分算法

基于粗分算法结果，将加权 RF 作为分类器，进行点云精分类分割，以解决数据不平衡导致的分类分割准确度不高的问题。

### 2.1 决策树选取

由于随机采样和特征随机选择的随机性，RF 中可能会包含一些性能较差的决策树，从而影响最终模型的预测精度，尤其在数据特征无关的情况下更加明显<sup>[18]</sup>。针对此类问题，以分类分割精度作为分类器性能衡量指标，计算传统 RF 中每棵决策树  $\{T_1, T_2, \dots, T_n\}$  的分类分割精度  $OA_i (i=1, 2, \dots, n)$ ，并据此对决策树进行排序，同时剔除分类分割精度低于平均分类分割精度的决策树，得到高精度的决策树  $\{T_1, T_2, \dots, T_u\}, (u < n)$ 。

RF 作为一种集成模型，尽管通常能提高分类分割性能和泛化能力，但如果内部的树结构或预测结果过于相似，会影响整体的泛化效果。针对此类问题，可采用多种度量方法来评估性能和泛化能力。在此采用不一致度量(Disagreement Measure, Dis)作为为决策树多样性衡量指标。

Dis 是测量 2 棵树的预测结果中有多少不一致，不一致的分类器能提升集成模型的多样性和性能，计算式为

$$\text{Dis}(C_i, C_j) = \frac{D_{01} + D_{10}}{D} \quad (6)$$

式中： $D_{01}$  表示分类器  $C_i$  预测错误而  $C_j$  预测正确的样本数； $D_{10}$  表示分类器  $C_i$  预测正确而  $C_j$  预测错误的样本数； $D$  表示总样本数量。

2 个分类器的 DIS 的取值范围是[0,1]。DIS 数值越大，2 个分类器之间的多样性程度就会越高；反之多样性程度就会越低。

利用式(1)对 RF 中任意 2 棵决策树间的 DIS 进

行计算，并置入矩阵中，组成不一致度量矩阵，即

$$\text{Dis} = \begin{bmatrix} \text{Dis}_{T_1, T_1} & \text{Dis}_{T_1, T_2} & \dots & \text{Dis}_{T_1, T_p} \\ \text{Dis}_{T_2, T_1} & \text{Dis}_{T_2, T_2} & \dots & \dots \\ \dots & \dots & \dots & \dots \\ \text{Dis}_{T_p, T_1} & \dots & \dots & \text{Dis}_{T_p, T_p} \end{bmatrix} \quad (7)$$

在计算出 RF 不一致度量矩阵后，通过设定不一致度量阈值  $\text{Dis}_{\text{Thresh}}$ ，将  $\text{Dis}_{T_i, T_j} < \text{Dis}_{\text{Thresh}}$  对应的决策树归为一类，最终将选取的  $u$  棵决策树  $\{T_1, T_2, \dots, T_u\}, (u < n)$  分为  $v$  个类。最后，从每个类别中挑选出分类分割精度最高的决策树，组成一个含有  $v$  个决策树的 RF 分类器  $\{\{T_1, T_2, \dots, T_u\}, (u < v < n)\}$ 。

### 2.2 加权参数选取

吴冬等<sup>[19]</sup>基于各个决策树的特征重要性与 RF 总体特征重要性的相关程度提出了一种加权投票方式，特征重要性评估主要分为基于基尼指数和基于袋外数据 2 种。在此基础上，提出动态权重优化与特征重采样相结合的新策略，使 RF 模型更加精准且具备自适应性。通过引入特征重要性与分类分割错误率的联合评估，以及动态调整权重与重采样机制，增强模型的性能。

首先，不仅基于袋外数据  $T_i$  计算每棵决策树的特征重要性，还计算其在袋外数据上的分类分割错误率  $E_i$ 。使用特征重要性和分类分割误差率的组合指标来动态调整权重，组合指标计算式为

$$W_i = \frac{\rho(H_i, H)}{E_i + \varepsilon} \quad (8)$$

式中： $W_i$  表示决策树  $T_i$  的权重； $\rho(H_i, H)$  表示决策树  $T_i$  的特征重要性与 RF 整体特征重要性  $H$  之间的皮尔逊相关系数； $E_i$  表示范围，指该决策树在袋外数据上的分类分割错误率； $\varepsilon$  表示防止分母为零的小量。

其次，在每轮训练中，适当调整弱特征的采样概率，并且更新采样权重，使模型不再过度依赖部分关键特征，避免过拟合问题。设特征  $f_j$  的重要性为  $I(f_j)$ ，则该特征的采样概率计算式为

$$P(f_j) = \frac{I(f_j)^\alpha}{\sum_{j=1}^d I(f_j)^\alpha} \quad (9)$$

式中： $\alpha \in [0, 1]$  表示调节参数，用于控制关键特征和弱特征之间的权衡。若  $\alpha$  接近 0，则所有特征的采样概率趋于平均；若  $\alpha$  接近 1，则更偏向于高重要性的特征。最后，为了进一步提升模型的性能，

引入了逐轮迭代机制。每轮迭代都基于上轮训练的结果, 动态调整决策树的权重和特征的采样策略, 确保模型不断趋于最优。若性能指标在连续几轮内不再提升, 则停止训练, 确保计算效率。

### 2.3 点云精分算法步骤

在精分算法部分, 经过决策树和加权参数的选取后, 分类分割步骤如下:

**步骤 1.** 假设经过点云粗分算法提取的特征向量为  $\mathbf{X}'' \in \mathbb{R}^d$ , 其中  $d$  表示特征维度。将提取到的特征向量  $\mathbf{X}''$  输入到每棵决策树。

**步骤 2.** 对于加权 RF 中的每棵决策树  $T_i$  (其中  $i=1, 2, \dots, N$ , 且  $N$  为决策树的总数), 树  $T_i$  会对输入特征  $\mathbf{X}''$  进行分类分割, 输出一个类别预测结果  $y_i$ 。通常,  $y_i$  为离散值, 表示预测的类别标签。为确保决策树多样性并增强模型泛化能力, 使用 Dis 来衡量 2 棵树预测结果的不一致性。

**步骤 3.** 给每棵决策树  $T_i$  分配了权重  $w_i$ , 并结合袋外数据的特征重要性和分类分割错误率来动态调整权重。

**步骤 4.** 针对每个类  $c_k \in \mathcal{C}$ , 得到类别得分为

$$S(c_k) = \sum_{i=1}^N w_i \cdot \zeta(y_i = c_k) \quad (10)$$

式中:  $\zeta(y_i = c_k)$  表示当  $y_i = c_k$  时为 1, 否则为 0。

为防止模型过度依赖部分特征, 引入特征重采样。通过逐轮迭代机制, 动态调整权重与采样策略, 使得模型逐步优化。

**步骤 5.** 最终预测类别  $y_{\text{final}}$  为得分最高的类别, 即

$$y_{\text{final}} = \text{avg} \max_{c_k \in \mathcal{C}} S(c_k) \quad (11)$$

经过精分算法后, 采用分类分割总体准确率 (Overall Accuracy, OA) 作为点云分类分割评价指标, 即

$$OA = \frac{TP + TN}{TP + TN + FP + FN} \quad (12)$$

式中:  $TP$  表示模型预测为正例且真实为正例的样本数;  $TN$  表示模型预测为负例且真实为负例的样本数;  $FP$  表示模型预测为正例但真实为负例的样本数;  $FN$  表示模型预测为负例但真实为正例的样本数。

## 3 实验结果与分析

### 3.1 实验数据与参数设置

实验采用 ModelNet40 数据集、Semantic3D 数

据集和室外场景数据作为算法有效性的测试数据。ModelNet40 是 Princeton 大学制作并公开的点云分类数据集, 包含了 Airplane, Cup, Flower Pot, Desk 和 Guitar 等 40 个类别的 12 311 个点云数据, 其中训练数据和测试数据分别有 9 843 个和 2 468 个, 如图 2 所示。在点云分割任务中, 室外场景数据包括房屋、树木、道路等类别, 如图 3 所示。Semantic3D 数据集与室外场景数据集类似, 同为室外场景点云, 包括建筑、道路、树木等(图 4)。参数设置为: 在点云粗分算法中, 模型训练的学习率为 0.001, Batch size 为 32, Epochs 为 250, 损失函数使用交叉熵损失, 激活函数为 ReLU, 优化器为 Adam; 在点云精分算法中, 不一致度量阈值为 0.6, 动态权重计算基于特征重要性和袋外数据错误率, 调节参数  $\alpha$  设为 0.5 以平衡关键特征与弱特征采样权重, 并在连续 5 轮无提升时提前终止训练。

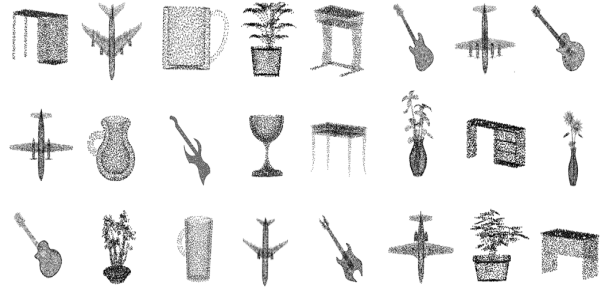


图 2 ModelNet40 部分点云数据

Fig. 2 Partial point cloud data from ModelNet40

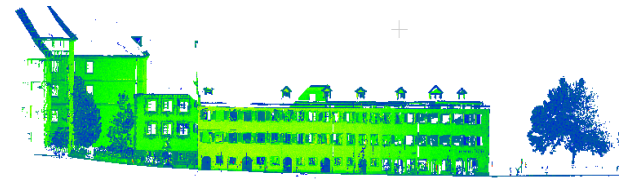


图 3 Semantic3D 部分点云数据

Fig. 3 Partial point cloud data for Semantic3D



图 4 部分室外场景点云数据

Fig. 4 Partial outdoor scene point cloud data

### 3.2 粗分算法中轻量级网络设计

规定网络中的各层之间相互独立。因此在实验

过程中,对于针对某一层参数优化实验时,保持其他层的参数设置与图 1 保持一致;通过对该层不同的参数取值进行多次实验,确定最佳的参数设置。具体步骤如下:

**步骤 1.** 通过不同层数卷积层分析轻量化特性。

以图 1 提供的初始框架为基础,设置 3 组对比实验:模型 A 为 2 层卷积,通道数 [32, 64];模型 B 为 3 层卷积,通道数 [16, 32, 64];模型 C 为 4 层卷积,通道数 [8, 16, 32, 64],希望通过增加卷积层的数量和特征维度,观察模型在分类分割任务中的表现,并分析不同架构下模型的参数量与性能的关系,以便找到精度和计算资源间的平衡点。所有实验通过多个逐点卷积层来提取特征,并最终映射到 40 个类别的输出。每个实验的训练过程会打印每个 Epoch 的准确率,Epoch 值为 250。

实验过程中模型 A 和 B 的学习速度较快,模型 C 因卷积层更多,速率方面表现较差;模型 B 在 250 轮训练后取得最高准确率 79.58%,表现优于其他模型;最终,模型 B 在计算效率与性能之间达到了最佳平衡,即 3 层卷积优于 2 层卷积和 4 层卷积。

**步骤 2.** 通过不同通道数的 3 层卷积分析轻量化特性。

在步骤 1 中,已经实验出 3 层卷积在计算效率与性能较优,但由于通道数设置为 64,导致模型的精度较差,现设置 A, B, C 共 3 组对比实验,在 3 层卷积的基础上,分别对应设置 128, 256 和 512 的通道数,并测试通道数对模型精度的影响。

实验结果中 3 个模型分类分割精度接近,但在模型参数方面,模型 A(通道数 128)的参数远小于模型 B(通道数 256)和模型 C(通道数 512)。因此,选择 3 层卷积的通道数为 128。

**步骤 3.** 通过不同层数的 MLP 分析轻量化特性。

设置模型 A, B 和 C 共 3 组对比实验:1 层 MLP,输出通道数为 128;2 层输出通道数为 128,第一层 128,第二层 64;3 层 MLP,输出通道数为 128,第一层 128,第二层 64,第三层 32。

实验中 3 组模型的精确度都达到了 92%,但是模型 A 的参数量最少,比模型 B 和 C 分别少了 2.36%和 2.68%,所以选择模型 A:1 层 MLP。

综上所述,由通道数为 128 的 3 层卷积以及通道数为 128 的 1 层 MLP 组成轻量级的神经网络精

度最佳,且时间效率最高。

### 3.3 精分算法精度分析

在 ModelNet40 数据集中设置 4 组对比实验:原始 RF、静态特征权重优化 RF、静态分类分割错误率优化 RF、动态联合评估权重优化 RF(本文算法)。为了量化特征重要性与分类分割错误率的联合评估对模型性能的贡献,使用弱特征贡献提升率进行评估,即

$$IR = \frac{P_{\text{enhanced}} - P_{\text{original}}}{P_{\text{original}}} \quad (13)$$

式中:  $P_{\text{enhanced}}$  表示使用原始特征加上增强或新增弱特征后的模型性能指标(如准确率、F1 分数等);  $P_{\text{original}}$  表示仅使用原始特征时的模型性能指标。

F1 分数计算式为

$$F1 = \frac{2 \times TP}{2 \times TP + FP + FN} \quad (14)$$

式中:  $TP$  表示模型预测为正例且真实为正例的样本数;  $FP$  表示模型预测为正例但真实为负例的样本数;  $FN$  表示模型预测为负例但真实为正例的样本数。

表 1 展示了 4 种实验模型在分类分割精度、F1 分数、弱特征贡献提升率对比结果。

表 1 不同实验模型在各性能指标中的对比

Table 1 Comparison of different experimental models in each performance index

模型	OA/%	F1	IR/%
原始 RF	85.30	83.9	—
静态特征权重优化 RF	87.10	85.5	15.6
静态分类分割错误率优化 RF	86.80	85.0	10.3
动态联合评估权重优化 RF	<b>89.96</b>	<b>88.1</b>	<b>32.8</b>

注:加粗数据表示最优值。

从表 1 可以看出,联合评估模型在分类分割性能(准确率和 F1 分数)上明显优于单一指标评估模型和基准随机森林,且联合评估模型能够通过动态调整权重和特征采样概率,提升弱特征的贡献率,显著增强模型的鲁棒性和泛化能力。

### 3.4 分类分割结果与对比分析

#### 3.4.1 点云分类实验结果

使用本文算法在 ModelNet40 数据集上进行分类,实验中训练数据和测试数据分别有 9 843 个和 2 468 个,其中 Airplane, Cup, Flower pot, Desk 和 Guitar 的分类结果如图 5 所示。

由图 5 可以看出,本文算法可准确地将 ModelNet40 的 40 个类别中的同一类标记为同一种

颜色, 如: Airplane 为蓝色、Cup 为黄色、Flower pot 为橙色、Desk 为绿色、Guitar 为紫色。

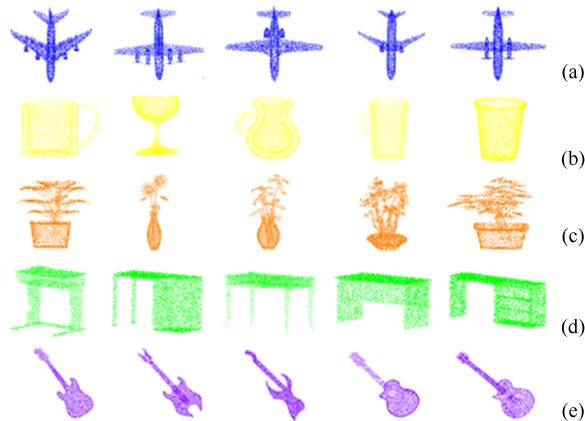


图 5 本文算法在 ModelNet40 数据集部分点云分类结果 ((a) 飞机; (b) 杯子; (c) 花盆; (d) 桌子; (e) 吉他)

**Fig. 5** Point cloud classification results of the proposed algorithm in part of the ModelNet40 dataset ((a) Airplane; (b) Cup; (c) Flower pot; (d) Desk; (e) Guitar)

### 3.4.2 点云分割实验结果

为了进一步验证本文算法的性能, 使用该算法在 Semantic3D 部分点云数据上进行分割, 结果如图 6 所示。

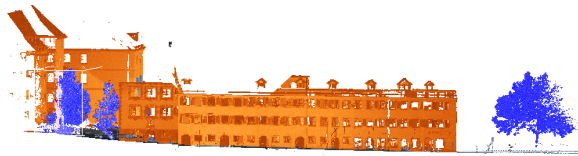


图 6 本文算法在 Semantic3D 部分点云数据点云分割结果  
**Fig. 6** Point cloud segmentation results of the proposed algorithm on partial point cloud data of Semantic3D

由图 6 所示, 本文算法将 Semantic3D 部分数据分成建筑(橘色)、树木(蓝色)、道路(黑色)、其他(深绿色)4 个部分。

使用本算法在室外场景原始点云数据上进行分割, 结果如图 7 所示。

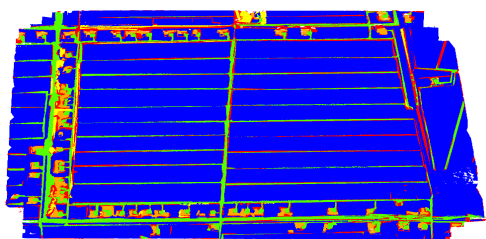


图 7 本文算法在室外场景原始点云数据点云分割结果  
**Fig. 7** Point cloud segmentation results of the proposed algorithm in the original point cloud data of outdoor scenes

由图 7 可以看出, 本文算法将室外场景点云分成田地(蓝色)、土地(绿色)、房屋(橙色)、树木(黄色)、其他(红色)5 部分。具体细节如图 8 所示。

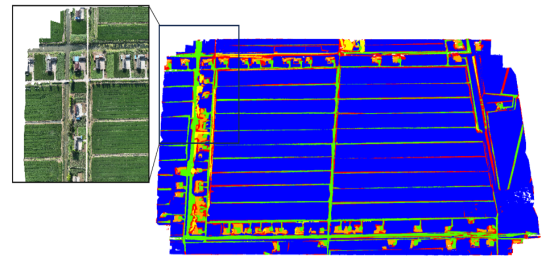


图 8 本文算法在室外场景原始点云数据点云分割结果细节

**Fig. 8** Details of the point cloud classification results of the proposed algorithm in the original point cloud data of outdoor scenes

从图 8 可以看出, 即使在室外场景原始点云数据类别多变、细节复杂的情况下, 本文算法依旧可以将每个部分区分出来。

### 3.4.3 点云分类分割实验结果分析

针对 ModelNet40, Semantic3D 部分数据集和室外场景点云数据, 采用 MVCNN, 3DshapeNets, VoxNet, PointNet, PointCNN, PointNet++, RG-GCN, Local Geo-Transformer, PointNeXt, FastPointNet++与本文算法进行对比实验。不同网络在 ModelNet40 数据集、Semantic3D 部分数据集和室外场景点云数据的分类分割结果见表 2。

由表 2 可以看出, 本文算法在 ModelNet40、Semantic3D 和室外场景点云数据集上均表现出显著优势。在 ModelNet40 上, 本文算法以 93.23% 的准确率先, 较次优的 PointNeXt(92.81%)提升 0.42%, 同时参数量( $2.36 \times 10^6$ )减少 5.60%, 训练时间(143 min)和单个物体识别时间(0.39 ms)均为最优。在 Semantic3D 上, 91.80% 的准确率配合  $2.50 \times 10^4$  的参数量, 以及 55 min 训练时间和 0.70 ms 识别速度, 展现了较为高效的性能。室外场景点云数据中, 91.24% 的准确率与  $3.51 \times 10^6$  的轻量参数量, 进一步验证了算法的适用性。

与之对比, MVCNN 基于多视角投影方法, 将点云数据投影为多个 2D 图像, 通过卷积神经网络进行处理, 虽然能够利用成熟的 2D 图像处理技术, 但投影过程会丢失三维几何信息, 网络参数量巨大, 训练时间长, 推理效率低, 难以满足实际需求。3DshapeNets 和 VoxNet 依赖体素化表示, 将点云离散化为规则的三维网格进行处理, 然而体素化方法

表 2 不同网络在不同数据集上的结果对比

Table 2 Comparison of results of different networks on different datasets

数据集	网络	参数量	OA/%	训练时间/min	单个物体识别时间/ms
ModelNet40	MVCNN <sup>[20]</sup>	1.280 0×10 <sup>8</sup>	91.10	500~800	8.40
	3DshapeNets <sup>[21]</sup>	3.400 0×10 <sup>7</sup>	78.52	1200	15.00
	VoxNet <sup>[22]</sup>	8.900 0×10 <sup>5</sup>	84.02	360~720	5.20
	PointNet <sup>[23]</sup>	3.540 0×10 <sup>6</sup>	88.53	254	0.82
	PointCNN <sup>[24]</sup>	7.850 0×10 <sup>5</sup>	89.32	600~800	1.50
	PointNet++ <sup>[7]</sup>	1.480 0×10 <sup>6</sup>	90.50	663	11.23
	RG-GCN <sup>[6]</sup>	2.200 0×10 <sup>6</sup>	91.00	700~900	2.50
	Local Geo-Transformer <sup>[9]</sup>	2.500 0×10 <sup>5</sup>	92.50	193	0.54
	PointNeXt <sup>[10]</sup>	4.200 0×10 <sup>5</sup>	92.81	180	0.46
	FastPointNet++ <sup>[11]</sup>	2.500 0×10 <sup>5</sup>	91.57	150	0.42
	本文算法	<b>2.359 4×10<sup>5</sup></b>	<b>93.23</b>	<b>143</b>	<b>0.39</b>
Semantic3D	MVCNN <sup>[20]</sup>	1.150 0×10 <sup>8</sup>	83.80	550~750	7.20
	3DshapeNets <sup>[22]</sup>	3.500 0×10 <sup>7</sup>	77.50	1100	17.00
	VoxNet <sup>[22]</sup>	1.200 0×10 <sup>6</sup>	81.80	270	5.40
	PointNet <sup>[23]</sup>	4.200 0×10 <sup>6</sup>	86.50	200	3.00
	PointCNN <sup>[24]</sup>	8.300 0×10 <sup>5</sup>	88.50	375~525	5.80
	PointNet++ <sup>[7]</sup>	1.450 0×10 <sup>6</sup>	89.90	640	8.30
	RG-GCN <sup>[6]</sup>	1.400 0×10 <sup>6</sup>	88.80	550~750	6.80
	Local Geo-Transformer <sup>[9]</sup>	5.200 0×10 <sup>5</sup>	90.10	90	1.40
	PointNeXt <sup>[10]</sup>	6.500 0×10 <sup>5</sup>	90.51	80	1.26
	FastPointNet++ <sup>[11]</sup>	3.800 0×10 <sup>5</sup>	89.83	65	0.90
	本文算法	<b>2.500 0×10<sup>5</sup></b>	<b>91.80</b>	<b>55</b>	<b>0.70</b>
室外场景点云数据	MVCNN <sup>[20]</sup>	1.100 0×10 <sup>11</sup>	84.56	1700~2400	11.40
	3DshapeNets <sup>[22]</sup>	3.500 0×10 <sup>9</sup>	78.21	2900	21.20
	VoxNet <sup>[22]</sup>	1.500 0×10 <sup>9</sup>	79.34	900	7.56
	PointNet <sup>[23]</sup>	3.000 0×10 <sup>8</sup>	84.62	820	5.60
	PointCNN <sup>[24]</sup>	8.520 0×10 <sup>7</sup>	87.24	950~1200	8.12
	PointNet++ <sup>[7]</sup>	1.200 0×10 <sup>9</sup>	88.27	1450	10.34
	RG-GCN <sup>[6]</sup>	2.100 0×10 <sup>8</sup>	88.43	1400~1600	8.23
	Local Geo-Transformer <sup>[9]</sup>	3.318 3×10 <sup>6</sup>	88.96	190	3.40
	PointNeXt <sup>[10]</sup>	3.200 0×10 <sup>6</sup>	89.11	145	2.30
	FastPointNet++ <sup>[11]</sup>	3.200 0×10 <sup>6</sup>	89.56	164	2.90
	本文算法	<b>3.012 3×10<sup>6</sup></b>	<b>92.24</b>	<b>90</b>	<b>1.46</b>

注：加粗数据表示最优值。

受到分辨率限制，易丢失点云细节，且三维卷积的计算复杂度高，导致资源消耗大，性能有限。PointNet 和 PointNet++虽然直接操作原始点云数据，通过 MLP 提取全局特征，但对局部几何关系建模不足，限制了分类分割性能。PointNet++虽通过引入层次化特征聚合改善了局部特征提取，但导致网络参数量和计算复杂度显著增加。RG-GCN 则通过图卷积网络和几何变换器捕获局部点云特征，性能较为出色，但依然未能做到轻量化，训练和推理效率不及所提算法。Local Geo-Transformer 实现了轻量化，但模型过于依赖局部几何信息，对全局特征的捕获能力不足，使得在复杂场景或需要综合局部与全局信息的任务中，分类分割性能略显不足。PointNeXt 虽通过残差 MLP 优化效率，但参数和训练时间显示其轻量化不足；FastPointNet++通过稀疏化加速，但 OA 偏低，验证了效率-精度的权衡困境。

相比之下，本文算法设计了轻量级网络结构，直接操作点云，并结合加权 RF 分类器，不仅精准建模点云的全局和局部特征，还避免了数据不平衡导致的信息损失。该算法以较小的参数量实现了更高的准确率和更快的识别速度，体现了算法在高效性和轻量化上的优势，适合资源受限的实际应用场景。

## 4 结论与展望

针对传统分类分割方法存在计算开销大、模型复杂度高、精度不够高等问题，提出了一种基于轻量级网络和加权随机森林的点云分类分割算法。通过轻量级网络高效提取点云特征，并结合不一致度量筛选与动态加权机制优化 RF，算法在分类分割精度和计算效率上均表现出显著优势。实验验证表明，该方法在 ModelNet40、Semantic3D 及室外场景点云数据上的性能优于现有主流算法，但对极端

稀疏或噪声密集的点云数据鲁棒性不足, 可能影响分割一致性。

未来工作将重点从以下 3 个方面展开改进与拓展:

1) 在算法层面引入自适应参数学习机制和基于注意力机制的动态加权策略, 减少人工调参依赖并提升模型对噪声和稀疏数据的鲁棒性;

2) 针对实时性需求场景优化计算流程, 探索模型压缩和硬件加速方案, 推动算法在实际工程中的部署应用;

3) 将方法扩展至三维目标检测、场景语义分割等关联任务, 验证其跨领域迁移能力, 同时开发面向大规模点云数据的分布式处理框架, 进一步提升算法的实用价值和应用广度。

### 参考文献 (References)

- [1] 贾明超, 冯斌, 吴鹏, 等. 一种融合改进 A\*算法与改进动态窗口法的文旅服务机器人路径规划[J]. 图学学报, 2024, 45(3): 505-515.  
JIA M C, FENG B, WU P, et al. A path planning for cultural tourism service robot combining improved A\* algorithm and improved dynamic window approach[J]. Journal of Graphics, 2024, 45(3): 505-515 (in Chinese).
- [2] 杨振, 李俊丽, 杨立炜, 等. 安全性 A\*融合 DWA 的分布式多移动机器人路径规划方法[J]. 控制工程, 2024, 31(12): 2284-2295.  
YANG Z, LI J L, YANG L W, et al. Distributed multi-mobile robots path planning method based on safe A\* fused with dynamic window approach[J]. Control Engineering of China, 2024, 31(12): 2284-2295 (in Chinese).
- [3] 成欢, 王硕, 李孟, 等. 面向自动驾驶场景的神经辐射场综述[J]. 图学学报, 2023, 44(6): 1091-1103.  
CHENG H, WANG S, LI M, et al. A review of neural radiance field for autonomous driving scene[J]. Journal of Graphics, 2023, 44(6): 1091-1103 (in Chinese).
- [4] MENG X H, LU X Y, YE H L, et al. A new self-augment CNN for 3D point cloud classification and segmentation[J]. International Journal of Machine Learning and Cybernetics, 2024, 15(3): 807-818.
- [5] WEN J J, MA J, ZHAO Y H, et al. Point cloud semantic segmentation based on local feature fusion and multilayer attention network[J]. IET Computer Vision, 2024, 18(3): 381-392.
- [6] ZENG Z Y, XU Y Y, XIE Z, et al. RG-GCN: a random graph based on graph convolution network for point cloud semantic segmentation[J]. Remote Sensing, 2022, 14(16): 4055.
- [7] LIU H, TIAN S H. Deep 3D point cloud classification and segmentation network based on GateNet[J]. The Visual Computer, 2024, 40(2): 971-981.
- [8] LI S, WANG M, WU J, et al. CloudDeepLabV3+: a lightweight ground-based cloud segmentation method based on multi-scale feature aggregation and multi-level attention feature enhancement[J]. International Journal of Remote Sensing, 2023, 44(15): 4836-4856.
- [9] JING Y X, SHEIL B, ACIKGOZ S. A lightweight transformer-based neural network for large-scale masonry arch bridge point cloud segmentation[J]. Computer-Aided Civil and Infrastructure Engineering, 2024, 39(16): 2427-2438.
- [10] QIAN G C, LI Y C, PENG H W, et al. PointNeXt: revisiting PointNet++ with improved training and scaling strategies[C]// The 36th International Conference on Neural Information Processing Systems. New York: Curran Associates Inc., 2022: 23192-23204.
- [11] CHEN X, LIU K, ZHANG Z. A PointNet-based CFAR detection method for radar target detection in sea clutter[J]. IEEE Geoscience and Remote Sensing Letters, 2024, 21: 3502305.
- [12] ALFIO V S, PEPE M, COSTANTINO D. The use of random forest for the classification of point cloud in urban scene[J]. International Journal of Engineering Trends and Technology, 2024, 72(3): 1-9.
- [13] ZHAO F Q, HUANG H, XIAO N N, et al. A point cloud segmentation algorithm based on multi-feature training and weighted random forest[J]. Measurement Science and Technology, 2025, 36(1): 015407.
- [14] MOHAMED M, MORSY S, EL-SHAZLY A. Improvement of 3D LiDAR point cloud classification of urban road environment based on random forest classifier[J]. Geocarto International, 2022, 37(27): 15604-15626.
- [15] FENG G Q, LI W L, ZHAO X L, et al. LessNet: lightweight and efficient semantic segmentation for large-scale point clouds[J]. IET Cyber-Systems and Robotics, 2022, 4(2): 107-115.
- [16] ZHAO X, WANG L M, ZHANG Y F, et al. A review of convolutional neural networks in computer vision[J]. Artificial Intelligence Review, 2024, 57(4): 99.
- [17] RÖSSIG A, PETKOVIC M. Advances in verification of ReLU neural networks[J]. Journal of Global Optimization, 2021, 81(1): 109-152.
- [18] JALAL N, MEHMOOD A, CHOI G S, et al. A novel improved random forest for text classification using feature ranking and optimal number of trees[J]. Journal of King Saud University-Computer and Information Sciences, 2022, 34(6): 2733-2742.
- [19] 吴冬, 阎卫东, 王井利. 基于特征重要性加权的随机森林点云分类研究[J]. 电子测量技术, 2023, 46(20): 120-127.  
WU D, YAN W D, WANG J L. Random forest point cloud classification algorithm based on feature importance weighting[J]. Electronic Measurement Technology, 2023, 46(20): 120-127 (in Chinese).
- [20] LIANG G, ZHAO X L, ZHAO J H, et al. MVCNN: a deep learning-based ocean-land waveform classification network for single-wavelength LiDAR bathymetry[J]. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, 2023, 16: 656-674.
- [21] MOHAMMADI F G, CHEN C, SHENAVARMASOULEH F, et al. 3D-model shapenet core classification using meta-semantic learning[EB/OL]. [2025-01-10]. <https://arxiv.org/abs/2205.15869>.
- [22] MSALLAM M, SYRYAMKIN V I. Comparison of classical and deep learning classifiers for 3D objects recognition[EB/OL]. [2025-01-10]. <https://elibrary.ru/wgdizx>.
- [23] SHCHENYAVSKAYA L A, GURA D A, DYACHENKO R A. Analysis of the PointNet neural network architecture[J]. Herald of Dagestan State Technical University Technical Sciences, 2024, 50(4): 158-165.
- [24] HELL M, BRANDMEIER M, BRIECHLE S, et al. Classification of tree species and standing dead trees with lidar point clouds using two deep neural networks: pointCNN and 3DmFV-net[J]. PFG—Journal of Photogrammetry, Remote Sensing and Geoinformation Science, 2022, 90(2): 103-121.