IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 28 November 2022, accepted 11 December 2022, date of publication 15 December 2022,
date of current version 23 December 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3229901

RESEARCH ARTICLE

Automatic Identification of Street Trees With Improved RandLA-Net and Accurate Calculation of Shading Area With Density-Based Iterative α-Shape

JUAN LEI[1], HONGWEI LI[1], SHAN ZHAO[1], YONGJI WANG[1], YIRUI JIANG[2], AND GE ZHU[1]
[1]School of Geo-Science and Technology, Zhengzhou University, Zhengzhou 450052, China
[2]School of Computer and Artificial Intelligence, Zhengzhou University, Zhengzhou 450052, China

Corresponding author: Hongwei Li (lhw29691518@zzu.edu.cn)

This work was supported in part by the "Key Technology of Intelligent Robot Space Perception" project of Key Program of National Natural Science Foundation of China under Grant 42130112, and in part by the "Research on Intelligent Identification and Extraction Method of Intelligent Urban Management Department/Event Based on Multi-Modal Data Fusion" project of Science and Technology Tackling Plan Program of Henan Province under Grant 222102320220.

ABSTRACT Street trees extraction and further acquisition of their property information is one of the current research hotspots, which can be applied in various urban management. For the processing task of automated identification of tree points from large-scale point clouds in urban road scenes, we propose an improved RandLA-Net network that takes into account both extraction accuracy and complexity. The method can eliminate the problem of insufficient feature extraction and more redundancy within the RandLA-Net model, and achieve efficient recognition of the street trees. As an application of street tree recognition results, we studied the accurate calculation method of street tree shading area and designed a density-based iterative α-shape algorithm. The method effectively solves the problems of large density differences among different point sets and overestimation of shading area caused by the existing convex hull algorithms' unrefined contour reconstruction, and improves the calculation accuracy of the canopy shading area. Experiments on the Paris-Lille-3D dataset show that the improved RandLA-Net network improves the IoU of three classes by 3.19%∼6.46% and reduces the model complexity by 1.4% compared to the original model. The density-based iterative α-shape algorithm achieves more refined contour reconstruction and solves the overestimation of shading area by 26% and 9% due to the other two convex hull algorithms. Its ability to effectively calculate accurate shading area was confirmed.

INDEX TERMS MLS, point clouds, deep learning, street tree extraction, shading area calculation, bilateral augmentation, improved RandLA-Net, density-based iterative α-shape.

I. INTRODUCTION

Street trees usually refer to trees planted on both sides of the road. As an important part of urban refinement management and the only living infrastructure in the city, they can block the strong direct sunlight in summer and play a shading and cooling role on the road space to alleviate the inconvenience of the public in the summer hot weather travel. The identification and extraction of street trees and further acquisition of tree morphological parameters is one of the current research hotspots, which can be applied in various urban management, such as providing data basis for tree inventory, urban green coverage survey, urban planning, etc., and then providing auxiliary support for urban managers to develop optimal urban street tree planning schemes [1]. However, due to the large number and dense distribution of street trees and the complex and diverse surrounding environment, it is important to explore efficient and convenient methods

The associate editor coordinating the review of this manuscript and approving it for publication was Kumaradevan Punithakumar.

for street tree extraction. Traditional methods mainly use electronic total station or RTK to measure the street trees, but there are disadvantages such as high labor intensity and low operational efficiency, and the collected data are incomplete and not accurate enough for in-depth analysis.

Mobile laser scanning (MLS) systems dynamically collects point clouds of ground objects on both sides of the road during vehicle travel. The LiDAR equipped on the MLS has strong penetration capability and is able to obtain high-density, millimeter-scale 3D spatial information in urban scenes, thus showing powerful 3D reconstruction capability. Compared to video detection technology, MLS can also provide additional information, such as intensity, reflected echoes, etc., which will be also useful for post-application [2], [3]. The system is able to obtain rich tree side information and features using the side-view scanning mode, thus demonstrating the great potential of extracting street trees from complex road environments. Studies have been conducted to calculate morphological parameters such as tree height, diameter at breast height, and crown width from tree point clouds [4, 5], but the irregularity and complexity of the spatial structure of tree canopies lead to less than optimal accuracy and efficiency of canopy factor measurements. Moreover, the experimental part of these research results is mostly based on the individual tree points. And the quantitative study of shading area is rarely the focus of research.

The initial idea of calculating the shading area of street trees originates from the experience of people traveling in the hot summer. In order to achieve this goal, two tasks need to be solved: the extraction of street trees based on urban MLS point clouds, and the accurate calculation of the shading area of street trees. For the first task, due to the uniqueness of urban MLS point clouds with uneven spatial distribution of points, missing data due to occlusion, diverse targets and large variation in scale, besides, trees overlap each other and have different morphologies, which poses a great challenge to the 3D accurate identification of street trees. According to the existing literature, the commonly used extraction methods can be broadly classified into three types based on clustering [6], [7], based on area growth [8], [9], and based on projection [10], [11], [12]. However, the automation and intelligence level of these methods is low, there is too much human-computer interaction, and the accuracy and stability of extraction need to be improved.

In recent years, significant progress has been made in semantic segmentation of urban MLS point clouds with the development of deep learning techniques, which has improved the decoding capability of point clouds and enabled the gradual maturation of methods for extracting specific types of objects from the original 3D point clouds, thus providing a new research perspective for automatic identification and extraction of street trees. However, because point clouds are scattered, disordered and irregularly distributed in 3D space, traditional convolutional neural networks (CNNs) are difficult to be applied directly [13]. Some scholars have tried to project 3D point clouds onto images with different viewpoints and then extract target features from the images with the help of CNN [14], [15], [16], [17], but the limited projection viewpoints inevitably lead to the loss of point clouds geometric and structural information. Some scholars have also voxelized point clouds and then extracted features by 3D CNN [18], [19], [20], [21], but the voxelization process significantly increases the computational time and memory cost. Obviously, the original point clouds can portray the geometric structure of the target more accurately and directly. PointNet [22] is a pioneering deep learning model that learns point-by-point features directly on the point clouds with the help of a shared multilayer perceptron(MLP), but each point learns features independently, ignoring the relative relationship with neighboring points and their features. In view of this, PointNet++ [23] proposes a local feature extraction module by using PointNet in overlapping local regions to obtain the fine geometric structure near each point. Then, recurrent neural networks(RNN) [24], [25], [26], [27], graph convolutional networks (GCN) [28], [29], [30], and kernel convolution [31], [32], [33], [34] have also been used to capture the local features of each centroid. However, the above methods are mainly applicable to simple structured targets or indoor scene point clouds, and are difficult to extend to more complex tasks of semantic segmentation for large-scale point clouds in urban scenes. Hu et al. [35] combined random sampling and local feature aggregation modules to form a RandLA-Net using a widely used encoder-decoder structure with jump connections, making the network memory and computationally efficient that is suitable for the task of processing large-scale point clouds. We leverage the achievements of this network in semantic segmentation to achieve the extraction of street trees from urban MLS point clouds.

For the accurate shading area calculation of the extracted street trees, the key lies in how to precisely construct the boundaries of the canopy projection points. Some scholars calculate the area of each slice with the help of convex hull algorithm after slicing the canopy by elevation in order to calculate tree volume. Weiheng Xu et al. [36] used the Graham scanning algorithm to achieve automatic extraction of the canopy projection area and compared it with the traditional manual measurement method. The results showed that the correlation between them were better. Gang Cheng et al. [37] used the improved α-shape algorithm to calculate the projected area of tree canopy slices and summed them to obtain the canopy volume. They proved with experimental results that the value of α is only determined by point clouds density and canopy shape, and has good generalization among different tree species. Inspired by these studies, we propose a high-precision calculation method for the shading area of street trees. Firstly, using urban MLS point clouds as the data source, we design a semantic segmentation network to extract the street tree points. Then, the density-based iterative α-shape algorithm is used to rapidly reconstruct the contour polygons of the canopy projection points in order to achieve the accurate calculation of the shading area of street trees.

II. METHOD

A. STREET TREES EXTRACTION METHOD BASED ON IMPROVED RANDLA-NET

We designed the network based on RandLA-Net. In order to improve the accuracy of the identification results and simplify the complexity of the model, a new local feature extraction module is designed in the coding stage. It mainly includes the farthest point sampling (FPS) and bilateral augmentation modules, which are used for down-sampling and feature aggregation of the original point clouds, respectively.

1) SPATIAL DOWN-SAMPLING BASED ON FPS

Spatial down-sampling is to reduce the number of points, and the input point clouds are continuously down-sampled in the network to save computational resources and memory overhead. RandLA-Net used random sampling (RS) to achieve high efficiency of the network but sacrifice the accuracy and stability. In this paper, we choose FPS instead of RS. FPS selects the next sampling point by comparing the distance between sampled points and un-sampled points in the geometric space, which can achieve uniform sampling of the original data (Figure 1). Compared with RS, FPS can take the coverage of small-scale and low-density targets into account, and greatly retains the geometric properties of 3D objects while streamlining the point set, which is conducive to improving the generalization ability of complex point sets.

In order to compensate for the loss of point features caused by down-sampling, the local spatial representation of each point after sampling is learned by feature extraction and mapped to a higher feature space layer by layer to expand the perceptual field to obtain higher-level and more abstract features. However, RandLA-Net simply concatenated the encoded spatial geometric information with the corresponding attribute features after down-sampling to obtain the fused features of the points, so it didn't fully utilize the given information and its ability to capture local features is limited.

2) BILATERAL AUGMENTATION

We design the bilateral augmentation module for extracting point clouds features of different resolutions obtained after down-sampling (Figure 2). To enhance the network's ability to learn local features, the module adds the encoding of relative semantic information and bilateral offsets to the original structure. Specifically, the module divides the input information into two aspects of attribute features: geometric information and semantic information. Given the advantages of an MLP that it can represent the features flexibly in a high-dimensional embedding space, empirically, we use a single-layer MLP to process the semantic information to obtain preliminary semantic information and then encode it with relative semantic features. In addition, biases learned from each other from the input bilateral information are added separately to enhance the local context of each point, which is then cascaded with other encoded features to obtain the fused features of the points. Such design can effectively enhance

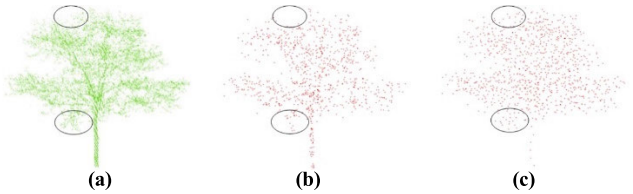


FIGURE 1. Comparison between two sampling methods. (a)original point sets; (b)point sets after RS; (c)point sets after FPS.

the network's ability to capture complex local features. Then multiple local feature extraction modules and attention pooling modules are stacked together to form a dilation residual block with the help of jump connections, and the perceptual field of each neural layer is gradually increased by convolution operations so that complex local features can be learned robustly and efficiently.

As shown in Figure 2, for a centroid p_i, the bilateral augmentation module first queries its K neighborhood points $\{p_i^1 \cdots p_i^k \cdots p_i^K\}$ and their corresponding features $\{f_i^1 \cdots f_i^k \cdots f_i^K\}$ with the help of the point-wise 3D-Euclidean distance-based K-nearest neighbors (KNN) algorithm. In order to capture both global and neighborhood local information so that the network can better learn the geometric structure of the space from the relative positions of the individual points as well as the distance information, the local geometric contexts $\mathcal{G}_\psi^k(p_i)$ are combined as:

$$\mathcal{G}_\psi^k(p_i) = MLP\left(p_i \oplus p_i^k \oplus \left(p_i - p_i^k\right) \oplus \| p_i - p_i^k \|\right) \tag{1}$$

where p_i and p_i^k are the three-dimensional coordinates of the points, $\oplus$ is the concatenation operation, and $\| \cdot \|$ calculates the Euclidean distance between the center point and its neighboring points.

Correspondingly, local semantic contexts $\mathcal{G}_\psi^k(f_i)$ can be encoded as:

$$\mathcal{G}_\psi^k(f_i) = MLP\left(f_i \oplus f_i^k \oplus \left(f_i - f_i^k\right)\right) \tag{2}$$

where f_i and f_i^k are the semantic features of the points.

Since $\mathcal{G}_\psi^k$ is strictly constructed under fixed constraints in three-dimensional space. Its generalization ability in high-dimensional feature space will be restricted. Furthermore, $\mathcal{G}_\psi^k$ neighbors may be redundant in representing closed regions. Thus, $\mathcal{G}_\psi^k(p_i)$ and $\mathcal{G}_\psi^k(f_i)$ may not be sufficient to represent neighbors. To address these issues, we add bilateral offsets to move the neighbors and associate them closely to the center of mass of the neighbors in order to augment the local context. First, an MLP is applied on the semantic context $\mathcal{G}_\psi^k(f_i)$ to estimate the 3-DoF (degree of freedom) bilateral offsets of the neighborhood. Thus, the shifted neighborhood points are:

$$\tilde{p}_i^k = MLP\left(\mathcal{G}_\psi^k(f_i)\right) + p_i^k, \tilde{p}_i^k \in \mathbb{R}^3 \tag{3}$$

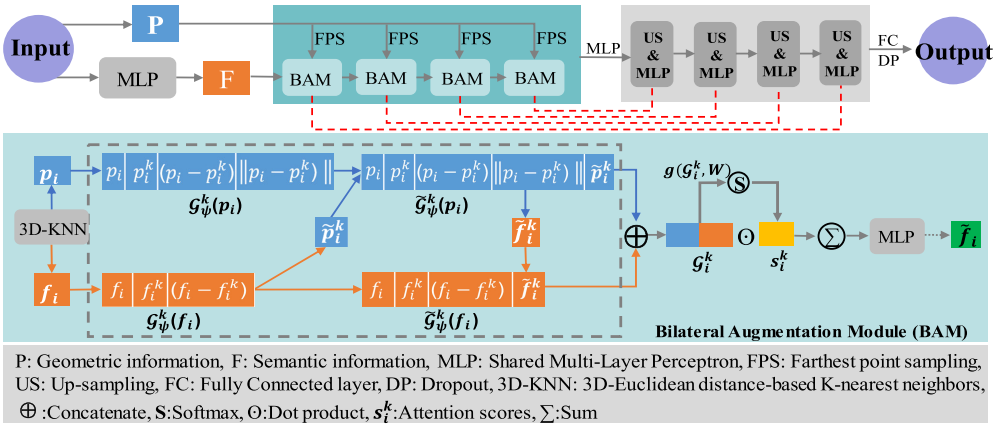


FIGURE 2. The structure of the improved RandLA-Net.

and the augmented local geometric context $\tilde{\mathcal{G}}_{\psi}^{k}(p_i)$ is:

$$\tilde{\mathcal{G}}_{\psi}^{k}(p_i) = MLP\left(p_i \oplus p_i^k \oplus \left(p_i - p_i^k\right) \oplus \| p_i - p_i^k \| \oplus \tilde{p}_i^k\right) \tag{4}$$

In order to make the augmented local geometric context further augment the local semantic context, the neighborhood point features $\tilde{f}_i^k$ are transformed as:

$$\tilde{f}_i^k = MLP\left(\tilde{\mathcal{G}}_{\psi}^{k}(p_i)\right) + f_i^k, \tilde{f}_i^k \in \mathbb{R}^d \tag{5}$$

and the augmented local semantic context $\tilde{\mathcal{G}}_{\psi}^{k}(f_i)$ is:

$$\tilde{\mathcal{G}}_{\psi}^{k}(f_i) = MLP\left(f_i \oplus f_i^k \oplus \left(f_i - f_i^k\right) \oplus \tilde{f}_i^k\right) \tag{6}$$

Finally, we concatenate the augmented local geometric context and the augmented local semantic context as an augmented local context $\mathcal{G}_{\psi}^{k}$:

$$\mathcal{G}_{\psi}^{k} = \tilde{G}_{\psi}^{k}(p_i) \oplus \tilde{\mathcal{G}}_{\psi}^{k}(f_i) \tag{7}$$

Then we learn a spatial mapping function $g()$, which maps K neighborhood positions of p_i in a continuous Euclidean space to obtain their corresponding mapping weight values. The mapping function g consists of a multilayer 1×1 convolution, batch normalization, and softmax. And W is the learnable weights of a shared MLP.

$$s_i^k = g\left(\tilde{\mathcal{G}}_{\psi}^{k}, W\right) \tag{8}$$

The features of these K neighborhood points and the corresponding weight values are then channel aggregated to finally generate aggregated features $\tilde{f}_i$ for p_i. This process is repeated several times to obtain higher-level features.

$$\tilde{f}_i = \sum_{k=1}^{K}\left(\tilde{\mathcal{G}}_{\psi}^{k} \cdot s_i^k\right) \tag{9}$$

B. SHADING AREA CALCULATION METHOD FOR STREET TREES BASED ON α-SHAPE CONVEX HULL ALGORITHM

1) SHADING AREA

The shading area is the area of shade produced on the ground by the canopy due to shading from direct sunlight, and we denote it by the projection of the canopy points on the ground. But the projection point set is scattered and disordered and does not have topological connections, which does not reflect the shape of the contour, and cannot be used directly to calculate the area. The traditional method is to consider the canopy projection as an ellipse and use the canopy amplitude values in more than two directions as the long and short axes of the ellipse. Then the canopy projection area can be calculated by the ellipse area formula [36], but the accuracy of this measurement method is less satisfactory.

2) α-SHAPE CONVEX HULL

In this paper, the canopy projection area is calculated using an α-shape convex hull polygon formed by the projection points. The α-shape algorithm is a boundary extraction and contour reconstruction for a finite and unordered set of points in the plane, and its concept was first proposed by EDELSBRUNNER et al. [38]. Other classical convex hull methods include Graham's scanning algorithm [39], Andrew's algorithm [40], and Jarvis step algorithm [41]. In two-dimensional space, α-shape algorithm can be imagined as a circle of radius α rolling through a finite and disordered set of points P. The value of α controls the fineness of the graph, and different α produces α-shapes of different morphology and topology, and it can be seen in Figure 3 that the smaller α is, the more closely the constructed convex hull fits the contours of the projected point set. But all α-shapes are the same for that point set is a subgraph of the Delaunay triangular dissection in Euclidean space. However, when the radius α tends to infinity, all the points in the point set P may be boundary points, which is unable to reconstruct the contour polygon. Theoretically, there exists a threshold α_b, and when $\alpha \geq \alpha_b$, the contour polygon formed by it can contain all the points in the point

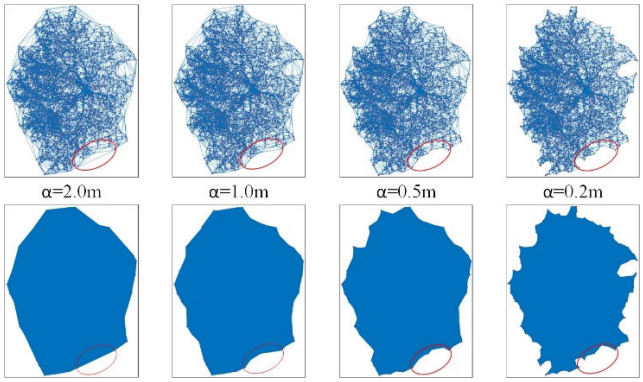


FIGURE 3. Boundary construction effect of point set at different α.

set P. Therefore the key of this algorithm is to select the appropriate global reconstruction factor α_b.

3) DENSITY-BASED ITERATIVE α–SHAPE

Since the point clouds acquired by different LiDAR scanning devices have large density differences, we proposed the density-based iterative α-shape algorithm for fast acquisition of the contour polygons of the projected point clouds. The key is to find the α_b value of each projected point set by iteration (Equation 8), so that the algorithm can take the density differences between different point clouds into account.

$$\alpha_{i+1} = \alpha_i + \Delta\alpha \ (i = 1, 2, \ldots, n) \tag{10}$$

where α_i, α_{i+1} are the radius of the circle at the i^{th}, $(i+1)^{th}$ iteration, $\Delta\alpha$ is the iteration step size, and i is the number of iterations.

The process of reconstructing the contour polygon and calculating the shading area for the canopy projection point clouds using our method is as follows.

(1) Obtain the projected point set. After denoising the canopy point clouds, we use the direct projection method to project all points along the Z-axis on the XOY plane.

(2) Construct the neighborhood point set and calculate its density. In order to improve the query speed, the overall spatial index structure of the projected point clouds is established with the help of KD-Tree, and the topological relationship of the local neighborhood is reconstructed to be able to quickly query the nearest point of each point and calculate the distance between two points. Then the average distance between all points is obtained as the density of this point set by cumulative summation and averaging.

(3) Iterative method to determine α_b and construct the convex hull. The initial value of α is set to 0, the iteration step $\Delta\alpha$ is set to point set density. The convex hull is constructed using the α-shape algorithm to determine whether only one regional convex hull is generated, and if not, the value of α is updated iteratively using Equation 8 until when only one convex hull is generated. Thus, the α_b of the point set is determined, and reconstruct the contour polygons under this threshold.

(4) Calculate the area of the canopy projection contour polygon. The area of the polygon can be accurately calculated by using the discretized Green's formula with the two-dimensional coordinates of the contour vertices (Equation 9), where m is the number of convex packet vertices and (x_i, y_i), (x_{i+1}, y_{i+1}) are the coordinate values of the i^{th} and $(i+1)^{th}$ points respectively.

$$S = \frac{1}{2}\sum_{i=1}^{m}\left[x_i\left(y_{i+1} - y_i\right) - y_i\left(x_{i+1} - x_i\right)\right] \tag{11}$$

C. EVALUATION METRICS

In order to evaluate the performance of semantic segmentation algorithms, accuracy is the most critical metric. The evaluation metrics in this paper mainly use the Intersection-over-Union (IoU) of each category and the mean value of IoUs (mIoU) for all semantic classes upon the whole dataset, and also provide the overall accuracy (OA) of all points to evaluate the experimental results.

$$IoU = \frac{T \cap P}{T \cup P} = \frac{TP}{TP + FP + FN} \tag{12}$$

$$mIoU = \frac{IoU}{m} \tag{13}$$

$$OA = \frac{\sum TP}{N} \tag{14}$$

where TP is the number of points in that category that are correctly predicted; FP is the number of points in other categories that are incorrectly predicted as that category; FN is the number of points in that category that are incorrectly predicted as other categories, N is the total number of points, and m is the total number of categories.

III. EXPERIMENT AND ANALYSIS

We divide the whole process into two parts, first extracting the tree points in the original point clouds with the deep learning semantic segmentation network based on RandLA-Net, and then calculating the shading area of single street trees separately after some refinement of the extracted street trees.

A. EXPERIMENTAL ENVIRONMENT AND DATASET

1) EXPERIMENTAL ENVIRONMENT

The experimental environment for model training and testing in this paper is Linux Ubuntu 18.4 operating system, Intel Core i9-10900K CPU, NVIDIA GeForce RTX 3090 GPU, 64GB and 24GB memory and video memory size respectively, implemented on Tensorflow deep learning framework. The network model hyper-parameters are set as follows: 100 epochs are trained on a GPU with a batch size between 4 to 8, depending on the amount of input points (about 36×2^{10} in Oakland 3-D to 64×2^{10} in Paris-Lille-3D). The Adam optimizer with default parameters is used to update the model, with momentum and initial learning rate set to 0.9 and 0.001, respectively, and the learning rate is the magnitude of the network update parameters, decaying at a rate of 0.5 every 10 epochs.

TABLE 1. Semantic segmentation results (%) on two datasets.

Dataset	IoU			mIoU	OA
	tree	sidewalk	others		
Paris-Lille-3D	88.25	92.63	93.36	91.42	95.84
Oakland 3-D	92.94	-	76.75	84.85	93.88

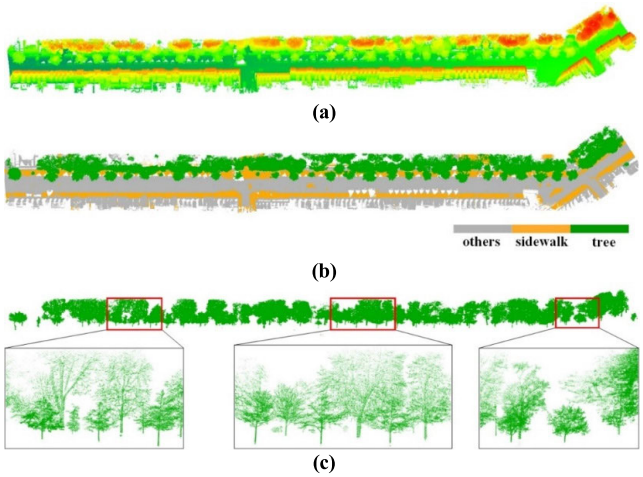


FIGURE 4. Visualization results of Paris-Lille-3D: (a) original point clouds; (b) semantic segmentation results and (c) tree points.

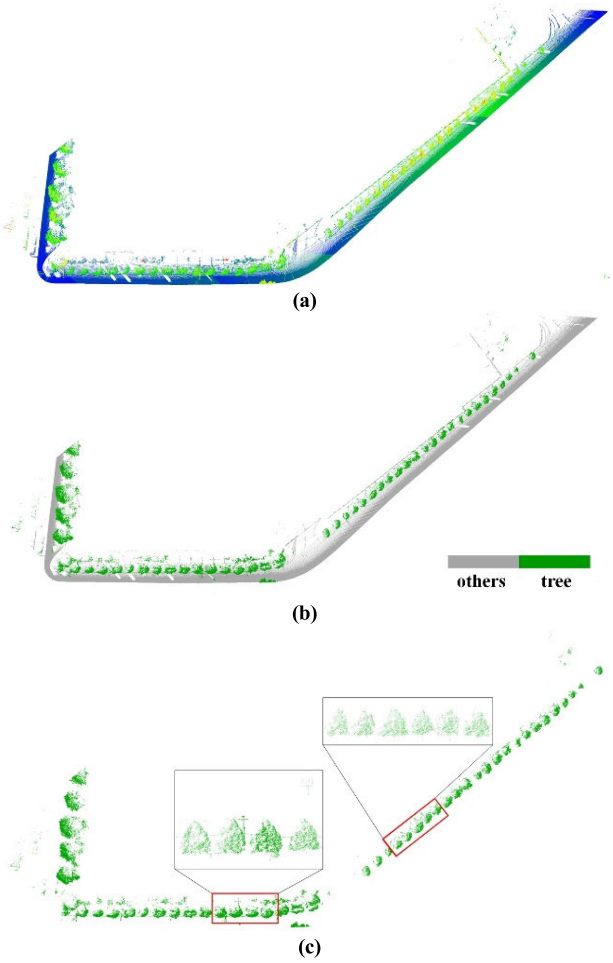


FIGURE 5. Visualization results of Oakland 3-D: (a)original point clouds; (b)semantic segmentation results; (c)tree points.

2) DATASET

To verify the effectiveness of our proposed approach, we conduct semantic segmentation experiments on two large-scale public point cloud datasets, both of which are derived from real-world scenarios in different countries.

Dataset I: The Paris-Lille-3D [42]public dataset contains point clouds of three outdoor scenes in Paris (2°19′ 49.31″∼2°19′ 57.98″E, 48°50′40.24∼48°50′55.35″N) and Lille (3°8′40.20″∼3°9′30.27″E, 50°40′16.15″∼ 50° 40′53.22″N). In this paper, Lille's point clouds were used as training data (about 100 million points) and Paris's point clouds were used as test data (about 45 million points), and three semantic categories of tree, sidewalk and others are defined.

Dataset II: The Oakland 3-D [43] dataset was collected around the CMU campus in Oakland, Pittsburgh, Pennsylvania. The dataset consists of 1.6 million points with 17 scene point clouds, with 10 scenes as training data (about 1 million points) and 7 scenes as test data (about 0.6 million points). Two semantic categories of tree and others are defined.

B. RESULTS OF STREET TREES EXTRACTION

1) STREET TREES EXTRACTION RESULTS BASED ON IMPROVED RANDLA-NET

Here are some details of the network implementation. First, we extract the tree points by the semantic segmentation deep learning network designed in this paper. Before training, a regular grid with the resolution of 0.04m is used to sample the entire scene to improve the computational efficiency of the network. In order to train the network in parallel, the input point clouds are continuously down sampled using CUDA-based farthest point sampling to save computational resources and memory overhead, with sampling ratios of 0.25, 0.25, 0.25, 0.25, and 0.5. In addition, we have enhanced the raw point clouds data in order to generate more training data during the training process. Specifically, the point clouds are fine-tuned so that each point is randomly perturbed in a small range in its original position, and then a rotation matrix around the vertical axis is added to the original geometric information for random rotation operations to increase the diversity of the input shapes. After training, in order to verify the generalization performance of our model, the network was tested using the model with the best mIoU after the training was completed. To quantitatively evaluate the performance of our model, IoU and OA were calculated for the categories in the Paris-Lille-3D and Oakland 3-D datasets, respectively, and Table 1 shows the performance of the corresponding quantitative evaluation metrics.

Figure 4 and Figure 5 show the visualization results of semantic segmentation using our algorithm on two datasets, Paris-Lille-3D and Oakland 3-D. It can be seen that the point

TABLE 2. Semantic segmentation results on the Paris-Lille-3D dataset.

Method	IoU(%) tree	sidewalk	others	mIoU(%)	OA(%)	Time(s)
PointNet	32.70	17.40	67.60	39.23	58.85	7.92
PointNet++	53.50	24.40	70.20	49.37	74.35	425.43
KPConv	**90.65**	85.59	87.49	87.91	94.19	29.57
RandLA-Net	83.46	86.17	90.17	86.60	93.61	7.61
BAAF-Net	87.97	91.08	92.28	90.44	95.38	13.59
Ours	88.25	**92.63**	**93.36**	**91.42**	**95.84**	12.74

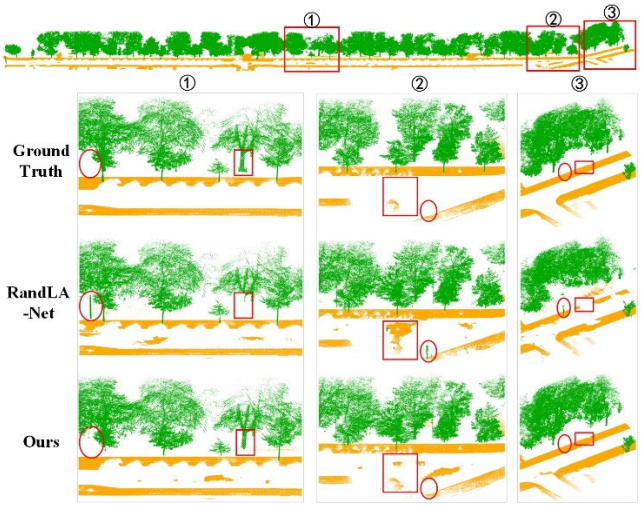


FIGURE 6. Local visual comparison between RandLA-Net and ours.

clouds collected on both sides of the road on the streets of Paris are only covered by street trees on one side, while the Oakland3-D dataset only collects point clouds on one side of the road. Compared with the Oakland3-D dataset, the Paris-Lille-3D point clouds are more dense, so the street trees have a clearer vertical structure and more obvious canopy features. From the results of local examples in Figure 4(c) and Figure 5(c), we can see that the overall performance of the semantic segmentation of the network designed in this paper is excellent. But there are also certain misclassifications, which are mainly because multiple objects overlap with each other and it is difficult to distinguish them. For example, the pole-like objects such as street lights and traffic signs erected in the trees, which have similar characteristics to tree trunks points leading to misclassification. In addition, there are also many power lines intertwined in the canopy in the Oakland 3-D dataset, and the power line point samples are small and not easy to identify. These problems have a large impact on the extraction results.

2) COMPARATIVE EXPERIMENT OF SEMANTIC SEGMENTATION

Table 2 compares the performance of our network with five other representative point clouds deep learning networks on the Paris-Lille-3D [42] dataset in terms of several quantitative evaluation metrics, namely the seminal model in point clouds deep learning PointNet [22] and its improved model PointNet++ [23], the state-of-the-art network KPConv [34] and the current approach BAAF-Net [44] and the baseline model RandLA-Net [35]. The inputs to all models are the x, y, z and reflection intensity information of the points. We record the average time in seconds for each model to process every 1 million points, and present the results in the last column of Table 2. The best metrics are bolded in Table 2, and it can be seen that the overall evaluation metrics of our method are better than those of other models, and the optimization effect is very obvious. PointNet++, as an improved model of PointNet, has a certain degree of improvement in all indicators, but its time cost per million points is 33 times higher than ours. The KPConv network achieves good results in extraction accuracy, but its spatial memory loss is more than 12.3 times than ours, which mainly because it relies on a complex kernel construction process. The algorithm proposed in this paper outperforms the RandLA-Net baseline model in all quantitative evaluation metrics, with IoU improvements of 3.19% to 6.46% for the three categories and an overall accuracy improvement of 2.23%. Figure 6 gives a comparison of the visualization effect between RandLA-Net and our method on the Paris-Lille-3D test dataset. It can be seen that the segmentation results of ours can improve the identification of some small-scale and low-density targets due to the use of FPS instead of RS, and the misclassification of pole-like objects can be effectively suppressed. Overall, thanks to the ability of our method to more fully learn the spatial features of the point clouds, our semantic segmentation results achieve a high degree of similarity compared with the real labels and is more satisfactory.

3) ABLATION STUDY

We conduct several ablation studies on Paris-Lille-3D [42], which evaluates the effectiveness of each module in terms of overall segmentation performance by controlling for variables.

In Table 3, A_1 inputs the x, y, z and reflection intensity features of the point clouds into the RandLA-Net [35] for training and uses it as the baseline model. A_2 shows that

TABLE 3. Ablation studies on the Paris-Lille-3D dataset.

Model		IoU(%) tree	sidewalk	others	mIoU(%)	OA(%)	Time(s)
A_1	RandLA-Net	83.46	86.17	90.17	86.60	93.61	7.61
A_2	Add bilateral augmentation module	84.03	87.69	90.59	87.44	93.94	7.62
A_3	Add normal vector features	86.35	81.89	89.63	85.96	93.37	7.62
A_4	Replace RS with FPS	88.25	92.63	93.36	91.42	95.84	12.74

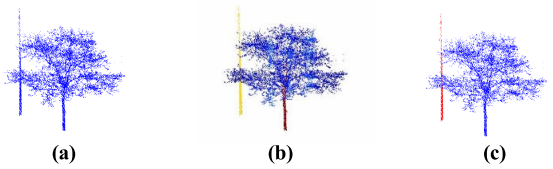


FIGURE 7. Projected point density method to remove pole-like objects: (a)original points; (b)clustering results in the vertical direction and (c)removal of pole-like objects.

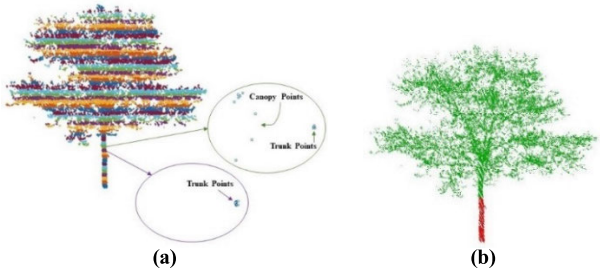


FIGURE 8. RANSAC slicing method to segment canopy and trunk points: (a)isometric slicing of tree points and (b)separating trunk points.

the addition of the bilateral augmentation module increases all the evaluation metrics slightly for each category, with the IoU improving by 0.57% and 1.52% for sidewalks and trees, respectively, and the average mIoU improving by 0.84%. Network complexity is crucial for practical applications of point clouds, and the spatial complexity is expressed in terms of the total number of parameters inside the model. The bilateral augmentation module proposed by us makes the total number of parameters required by the model change from 4992795 to 4923752, a reduction of 1.4%. It indicates that the bilateral augmentation module can not only improve the accuracy of point clouds target identification, but also effectively reduce the spatial complexity of the model. This is because more similar features are generated when features are mapped layer by layer to the high-dimensional space, and RandLA-Net fuses these similar features several times in the process of aggregating local features to enhance the perceptual capability of the model. In fact, this process causes information redundancy within the model and increases the complexity of the model to handle large-scale point clouds. We designed the bilateral augmentation module to be able to give some small offsets to the features according to their geometric positions, and when these features consider the information of geometry, their similarity will be reduced and the redundancy will be reduced. So after adding the bilateral augmentation module, the required parameters inside the model will be reduced and the spatial complexity will be decreased.

In order to further utilize the point clouds spatial information and obtain deep semantic information, A_3 adds normal vector features to the bilateral augmentation module and fuses them with the local feature aggregation module in the network, resulting in a 2.32% increase in tree extraction accuracy. A_4 shows that replacing RS with FPS in the down-sampling can effectively improve the accuracy of each category, in which the sidewalk IoU is improved by 10.74% and the average IoU of each category improves by 5.46%. Due to the CUDA-based FPS algorithm, the model running time is only 1.67 times longer than RS, and only 49 additional parameters are required. In summary, several modules designed in this paper have improved the overall performance of the model and proved their effectiveness.

C. STREET TREE SHADING AREA CALCULATION

1) CALCULATION OF STREET TREE SHADING AREA BASED ON α-SHAPE CONVEX HULL ALGORITHM

After extracting the tree points by our semantic segmentation network, it can be seen that the distribution of street trees in the datasets is sparse and there is little overlap. So we can directly use the Euclidean clustering [45] on these street tree points to split them into single trees. Euclidean clustering is a common method of partitioning point clouds into separate clusters based on the Euclidean distance metric.

Then, in order to improve the calculation accuracy of the shading area of the street trees, some refinement post-processing operations are also needed for the segmented individual street trees. Firstly, as shown in Figure 7(a), the pole-like objects distributed around the street trees such as street lights and traffic signs which are easily misclassified into tree points need to be cleared. Considering the structural characteristics of rods in the vertical direction, we perform density-based spatial clustering of applications with noise(DBSCAN) clustering in vertical direction to extract the rods (Figure 7(b)). DBSCAN is a density-based clustering algorithm, i.e., clustering is accomplished by differences in points density, where the points density is determined by counting the number of points in a specified radius area around the point. R. Ferrara et al. [46] demonstrated that the DBSCAN-based algorithm can effectively discover high-density regions and

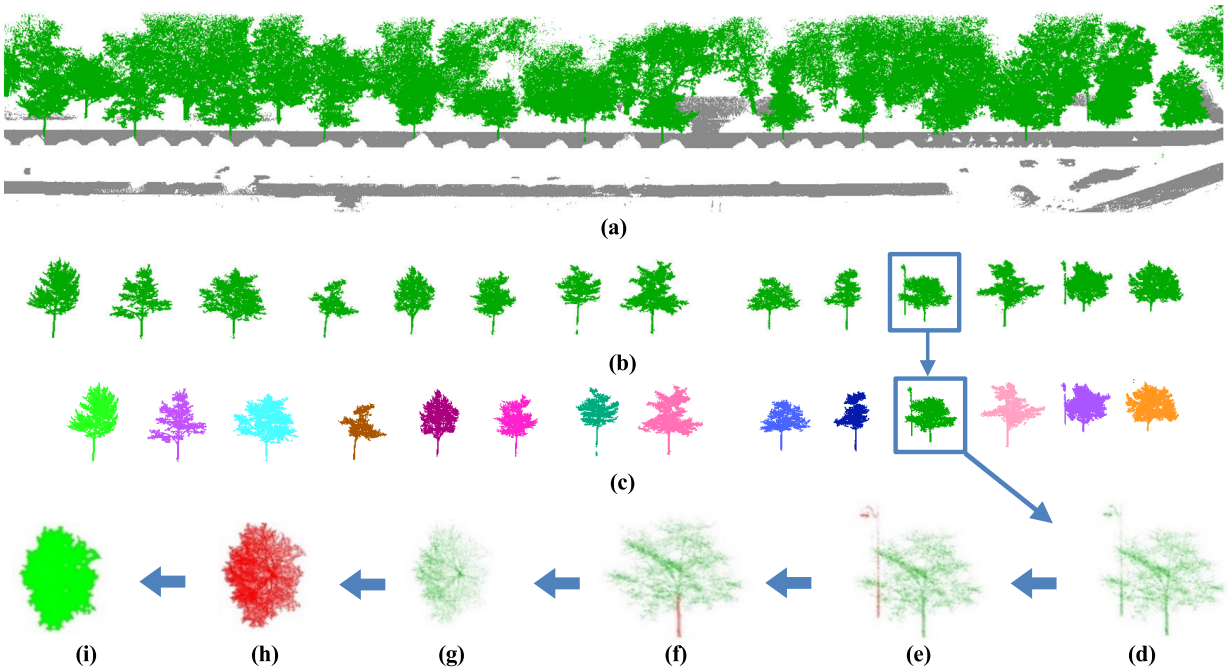


FIGURE 9. Shading area calculation process: (a)tree and sidewalk points; (b)street tree points; (c)single tree segmentation; (d)single street tree; (e)removing pole-like objects; (f)separating trunk points; (g)projection of canopy points; (h)constructing the convex hull and (i)calculating area.

group them into clusters. It is suitable for distinguishing pole-like objects and tree canopies with large differences in density in the vertical direction. However, the application to point clouds suffers from computational inefficiency, so we construct a KD-Tree for the input point cloud to reduce the computational complexity.

In order to obtain pure canopy points, the points of the trunk part need to be removed in order to calculate the canopy projection area. We use the slicing method based on random sample consensus (RANSAC) to separate the canopy and trunk, whose main idea is to stratify the original tree points with equal distance in the axial direction of elevation. As shown in Figure 8, the trunk points were removed by comparing the horizontal and vertical distance of two adjacent slices. The overall process is shown in Figure 9, and canopy shading area of street trees can be obtained after those steps.

For the same plant, the area of the canopy projection on the ground varies with the solar altitude angle during the day. In this paper, we only discuss the shading area produced by the solar radiation irradiating vertically on the ground, which is represented by the area of the canopy points projected vertically on the ground. We use the density-based iterative α-shape algorithm to reconstruct the contour of the canopy projection point set, which can take the density difference between the point sets into account, and then use the discretized Green's formula to calculate the area of the convex hull polygon.

The effective measurement distance of MLS used in the Paris-Lille-3D dataset is about 20 meters, which can cover the ground objects about 20 meters away from the centerline

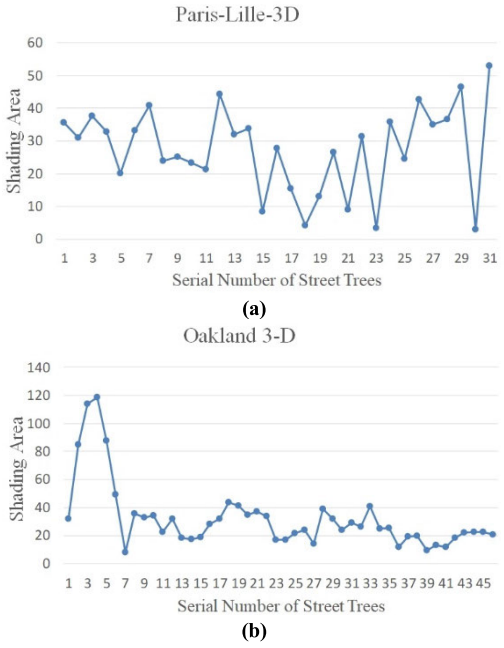


FIGURE 10. Statistical results of shading area: (a)Paris-Lille-3D dataset and (b)Oakland 3-D dataset.

of the road. Therefore, the collected point clouds contain the points of other trees outside the sidewalk, it is necessary to crop the street trees according to the location of the sidewalk in this area (Figure 9(a)). After the crop is finished, Figure 10(a) shows the street in Paris of Paris-Lille-3D dataset has 31 street trees on the 405-meter single-side sidewalk, with

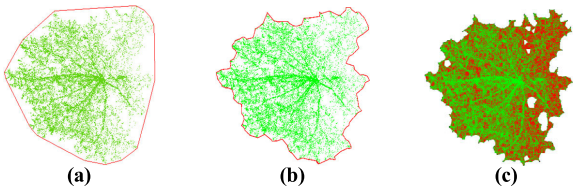


FIGURE 11. Contour reconstruction results of different convex hull algorithms: (a)Graham; (b)improved α-shape and (c)ours.

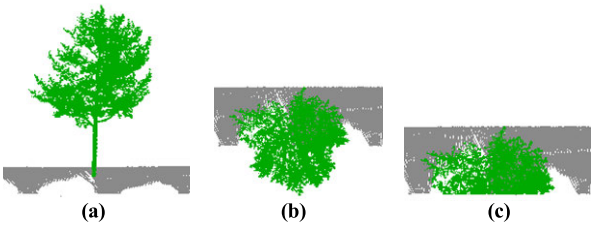


FIGURE 12. Calculation of sidewalk shading rate: (a) side-view of street trees; (b) top view of street tree and (c) sidewalk shading ratio.

a total shading area of 851.93 m^2, and the average shading area of a single street tree is 27.48 m^2. In the 475-meter road of Oakland 3-D campus scene, there are 46 street trees with a total shading area of 1486 m^2 (Figure 10(b)), and the average shading area of a single street tree is 32.30 m^2. The Oakland 3-D dataset has a slightly larger canopy per tree on average.

2) DIFFERENT SHADING AREA CALCULATION METHODS FOR STREET TREES

Figure 11(a), (b), and (c) show the effects of the Graham scanning algorithm [36], the improved α-shape algorithm [37], and our algorithm for contour reconstruction of the projected point set, respectively. It can be seen that the contour reconstructed by the density-based iterative α-shape algorithm are closer to the real situation, can better fit the shape of the canopy point clouds. The shading areas calculated by other two convex hulls are 29.52 m^2, 25.55 m^2, respectively, while our algorithm gives an area of 23.40 m^2. It can solve the overestimation of shading area by 26% and 9% due to the two convex hulls. Thus our method can obtain a more accurate shading area.

3) SIDEWALK SHADING RATIO CALCULATION

After calculating the shading area of a single street tree, some other parameters can be derived from this, such as sidewalk shading ratio, which is the percentage of the total area shaded by the vertical projection of the tree canopy on the sidewalk compared to the total area of the sidewalk. To achieve this goal, it is necessary to obtain the points of tree crowns projected on the sidewalk and calculate the total area of the sidewalk. Since MLS uses side-view scanning, the motor vehicles parked on the inner side of the sidewalk have a large impact on the data acquisition, which will form a triangle-like data void on the sidewalk behind the vehicles (Figure 12), resulting in the area of the sidewalk cannot be accurately obtained. So we calculate the area of the sidewalk according to the regular geometry, and gets the sidewalk width of Paris street is 4 meters by CloudCompare, the area is 1620 m^2. The shading area on the sidewalk is 545.49 m^2, and the shading ratio of the sidewalk is 33.67%. While the sidewalk width of Oakland 3-D dataset is 6 meters, the shading ratio is 52.14%. As can be seen, compared with the point clouds of the urban road scene collected by Paris-Lille-3D, the Oakland 3-D dataset, which collected the road in the campus point clouds, has a higher degree of greenery.

Since the distribution of street trees in the dataset selected in this paper is sparse. So our method may not be applicable to shading area calculation for complex cases such as multiple trees with spatial occlusion and overlapping, and we will focus on how to automate the separation of connected street trees in the future.

IV. CONCLUSION

In this paper, we proposed a complete automated process for extracting street trees from the urban large-scale scene point clouds obtained by MLS and then calculating their shading area. The first is that we designed an improved RandLA-net network for automated extraction of tree points, which improved the IoU of three categories by 3.19%∼6.46% and reduced the model spatial complexity by 1.4% compared to the original model. We also demonstrated the effectiveness of the method by performing ablation and comparison experiments.

Then, individual canopy points were obtained after the process of single tree segmentation, rod object rejection, and trunk separation. For calculating the canopy shading area, the density-based iterative α-shape algorithm was able to better fit the canopy projection contour and solve the overestimation of shading area by 26% and 9% than other two convex hull algorithms. It also enabled the calculation of sidewalk shading rate on this basis.

In conclusion, we believe that our contribution provides new insights in terms of automated tree extraction from urban MLS point clouds and further application to the scientific management of street trees.

ACKNOWLEDGMENT

The authors sincerely thank the anonymous reviewers for the critical comments and suggestions for improving the manuscript.

REFERENCES

[1] K. Anderson, S. Hancock, S. Casalegno, A. Griffiths, D. Griffiths, F. Sargent, J. McCallum, D. T. C. Cox, and K. J. Gaston, "Visualising the urban green volume: Exploring LiDAR voxels with tangible technologies and virtual models," *Landscape Urban Planning*, vol. 178, pp. 248–260, Oct. 2018, doi: 10.1016/j.landurbplan.2018.05.024.

[2] S. Luo, C. Wang, X. Xi, F. Pan, D. Peng, J. Zou, S. Nie, and H. Qin, "Fusion of airborne LiDAR data and hyperspectral imagery for aboveground and belowground forest biomass estimation," *Ecol. Indicators*, vol. 73, pp. 378–387, Feb. 2017, doi: 10.1016/j.ecolind.2016.10.001.

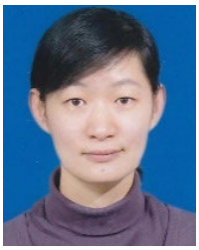
[3] F. T. Kurdi, M. Awrangjeb, and N. Munir, "Automatic filtering and 2D modeling of airborne laser scanning building point cloud," *Trans. GIS*, vol. 25, no. 1, pp. 164–188, Feb. 2021, doi: 10.1111/tgis.12685.

[4] D. R. Unger, I.-K. Hung, R. Brooks, and H. Williams, "Estimating number of trees, tree height and crown width using lidar data," *GISci. Remote Sens.*, vol. 51, no. 3, pp. 227–238, May 2014, doi: 10.1080/15481603.2014.909107.
[5] G. S. Maan, C. K. Singh, M. K. Singh, and B. Nagarajan, "Tree species biomass and carbon stock measurement using ground based-LiDAR," *Geocarto Int.*, vol. 30, no. 3, pp. 293–310, Mar. 2015, doi: 10.1080/10106049.2014.925003.
[6] S. Xu, S. Xu, N. Ye, and F. Zhu, "Automatic extraction of street trees' non-photosynthetic components from MLS data," *Int. J. Appl. Earth Observ. Geoinf.*, vol. 69, pp. 64–77, Jul. 2018, doi: 10.1016/j.jag.2018.02.016.
[7] Y. Xiao, S. Hu, S. Xiao, and A. Zhang, "A fast statistical method of tree information from 3D laser point clouds," *Chin. J. Lasers*, vol. 45, no. 5, 2018, Art. no. 0510007.
[8] L. Li, D. Li, H. Zhu, and Y. Li, "A dual growing method for the automatic extraction of individual trees from mobile laser scanning data," *ISPRS J. Photogramm. Remote Sens.*, vol. 120, pp. 37–52, Oct. 2016, doi: 10.1016/j.isprsjprs.2016.07.009.
[9] P. Babahajiani, L. Fan, J.-K. Kämäräinen, and M. Gabbouj, "Urban 3D segmentation and modelling from street view images and LiDAR point clouds," *Mach. Vis. Appl.*, vol. 28, no. 7, pp. 679–694, Oct. 2017, doi: 10.1007/s00138-017-0845-3.
[10] B. Wu, B. Yu, W. Yue, W. Tan, C. Hu, and J. Wu, "Method for identifying individual street trees from the cloud data of the vehicle-borne laser scanning points," *J. East China Normal Univ.*, vol. 2013, no. 2, pp. 38–49, 2013.
[11] S. Xu, S. Xu, N. Ye, and F. Zhu, "Individual stem detection in residential environments with MLS data," *Remote Sens. Lett.*, vol. 9, no. 1, pp. 51–60, Jan. 2018, doi: 10.1080/2150704x.2017.1384588.
[12] J. Zhou, G. Zhou, H. Wei, and X. Zhang, "Estimation of the plot-level forest parameters from terrestrial laser scanning data," in *Proc. IEEE Int. Geosci. Remote Sens. Symp. (IGARSS)*, Valencia, Spain, Jul. 2018, pp. 9014–9017.
[13] H. Fang and F. Lafarge, "Pyramid scene parsing network in 3D: Improving semantic segmentation of point clouds with multi-scale contextual information," *ISPRS J. Photogramm. Remote Sens.*, vol. 154, pp. 246–258, Aug. 2019, doi: 10.1016/j.isprsjprs.2019.06.010.
[14] F. J. Lawin, M. Danelljan, P. Tosteberg, G. Bhat, F. S. Khan, and M. Felsberg, "Deep projective 3D semantic segmentation," in *Proc. 17th Int. Conf. Comput. Anal. Images Patterns (CAIP)*, Ystad, Sweden, 2017, pp. 95–107.
[15] X. Chen, H. Ma, J. Wan, B. Li, and T. Xia, "Multi-view 3D object detection network for autonomous driving," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Honolulu, HI, USA, Jul. 2017, pp. 6526–6534.
[16] A. H. Lang, S. Vora, H. Caesar, L. Zhou, J. Yang, and O. Beijbom, "PointPillars: Fast encoders for object detection from point clouds," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Long Beach, CA, USA, Jun. 2019, pp. 12689–12697.
[17] B. Yang, W. Luo, and R. Urtasun, "PIXOR: Real-time 3D object detection from point clouds," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Salt Lake City, UT, USA, Jun. 2018, pp. 7652–7660.
[18] D. Maturana and S. Scherer, "VoxNet: A 3D convolutional neural network for real-time object recognition," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Sep. 2015, pp. 922–928.
[19] Y. Ben-Shabat, M. Lindenbaum, and A. Fischer, "3DmFV: Three-dimensional point cloud classification in real-time using convolutional neural networks," *IEEE Robot. Autom. Lett.*, vol. 3, no. 4, pp. 3145–3152, Oct. 2018, doi: 10.1109/LRA.2018.2850061.
[20] B. Graham, M. Engelcke, and L. van der Maaten, "3D semantic segmentation with submanifold sparse convolutional networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Salt Lake City, UT, USA, Jun. 2018, pp. 9224–9232.
[21] H.-Y. Meng, L. Gao, Y.-K. Lai, and D. Manocha, "VV-Net: Voxel VAE net with group convolutions for point cloud segmentation," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 8499–8507, doi: 10.1109/iccv.2019.00859.
[22] R. Q. Charles, H. Su, M. Kaichun, and L. J. Guibas, "PointNet: Deep learning on point sets for 3D classification and segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Honolulu, HI, USA, Jul. 2017, pp. 77–85.
[23] C. R. Qi, L. Yi, H. Su, and L. J. Guibas, "PointNet++: Deep hierarchical feature learning on point sets in a metric space," in *Proc. Adv. Neural Inf. Process. Syst.*, Long Beach, CA, USA, 2017, pp. 1–10.
[24] E. Shelhamer, J. Long, and T. Darrell, "Fully convolutional networks for semantic segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 4, pp. 640–651, Apr. 2017, doi: 10.1109/TPAMI.2016.2572683.
[25] Q. Huang, W. Wang, and U. Neumann, "Recurrent slice networks for 3D segmentation of point clouds," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Salt Lake City, UT, USA, Jun. 2018, pp. 2626–2635.
[26] Y. Xiaoqing, L. Jiamao, H. Hexiao, D. Liang, and Z. Xiaolin, "3D recurrent neural networks with context fusion for point cloud semantic segmentation," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 415–430, doi: 10.1007/978-3-030-01234-2_25.
[27] F. Engelmann, T. Kontogianni, A. Hermans, and B. Leibe, "Exploring spatial context for 3D semantic segmentation of point clouds," in *Proc. IEEE Int. Conf. Comput. Vis. Workshops (ICCVW)*, Venice, Italy, Oct. 2017, pp. 716–724.
[28] L. Landrieu and M. Simonovsky, "Large-scale point cloud semantic segmentation with superpoint graphs," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Salt Lake City, UT, USA, Jun. 2018, pp. 4558–4567.
[29] Y. Wang, Y. B. Sun, Z. W. Liu, S. E. Sarma, M. M. Bronstein, and J. M. Solomon, "Dynamic graph CNN for learning on point clouds," *ACM Trans. Graph.*, vol. 38, no. 5, p. 12, Nov. 2019, doi: 10.1145/3326362.
[30] L. Jiang, H. Zhao, S. Liu, X. Shen, C.-W. Fu, and J. Jia, "Hierarchical point-edge interaction network for point cloud semantic segmentation," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 10432–10440, doi: 10.1109/ICCV.2019.01053.
[31] P. Hermosilla, T. Ritschel, P. P. Vazquez, A. Vinacua, and T. Ropinski, "Monte Carlo convolution for learning on non-uniformly sampled point clouds," in *Proc. 11th ACM SIGGRAPH Conf. Exhib. Comput. Graph. Interact. Techn. Asia (SA)*, Tokyo, Japan, 2018, pp. 1–12.
[32] W. Wu, Z. Qi, and L. Fuxin, "PointConv: Deep convolutional networks on 3D point clouds," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 9621–9630.
[33] Y. Xu, T. Fan, M. Xu, L. Zeng, and Y. Qiao, "SpiderCNN: Deep learning on point sets with parameterized convolutional filters," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 90–105, doi: 10.1007/978-3-030-01237-3_6.
[34] H. Thomas, C. R. Qi, J.-E. Deschaud, B. Marcotegui, F. Goulette, and L. Guibas, "KPConv: Flexible and deformable convolution for point clouds," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Seoul, South Korea, Oct. 2019, pp. 6420–6429.
[35] Q. Hu, B. Yang, L. Xie, S. Rosa, Y. Guo, Z. Wang, N. Trigoni, and A. Markham, "RandLA-Net: Efficient semantic segmentation of large-scale point clouds," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 11105–11114.
[36] W. H. Xu, Z. K. Feng, Z. F. Su, H. Xu, Y. Q. Jiao, and O. Deng, "An automatic extraction algorithm for individual tree crown projection area and volume based on 3D point cloud data," *Spectrosc. Spectr. Anal.*, vol. 34, no. 2, pp. 465–471, Feb. 2014, doi: 10.3964/j.issn.1000-0593(2014)02-0465-07.
[37] G. Cheng, J. Wang, J. Yang, Z. Zhao, and L. Wang, "Calculation method of 3D point cloud canopy volume based on improved a-shape algorithm," *Trans. Chin. Soc. Agricult. Machinery*, vol. 52, no. 5, pp. 175–183, 2021, doi: 10.6041/j.issn.1000-1298.2021.05.019.
[38] H. Edelsbrunner, D. G. Kirkpatrick, and R. Seidel, "On the shape of a set of points in the plane," *IEEE Trans. Inf. Theory*, vol. IT-29, no. 4, pp. 551–559, Jul. 1983, doi: 10.1109/TIT.1983.1056714.
[39] R. L. Graham, "An efficient algorithm for determining the convex hull of a finite planar set," *Inf. Process. Lett.*, vol. 1, no. 4, pp. 132–133, Jan. 1972, doi: 10.1016/0020-0190(72)90045-2.
[40] A. Andrew and A. Am, "Another efficient algorithm for convex hulls in two dimensions," *Inf. Process. Lett.*, vol. 9, no. 5, pp. 216–219, 1979, doi: 10.1016/0020-0190(79)90072-3.
[41] R. A. Jarvis, "On the identification of the convex hull of a finite set of points in the plane," *Inf. Process. Lett.*, vol. 2, no. 1, pp. 18–21, Mar. 1973, doi: 10.1016/0020-0190(73)90020-3.
[42] X. Roynard, J.-E. Deschaud, and F. Goulette, "Paris-Lille-3D: A large and high-quality ground-truth urban point cloud dataset for automatic segmentation and classification," *Int. J. Robot Res.*, vol. 37, no. 6, pp. 545–557, May 2018, doi: 10.1177/0278364918767506.
[43] D. Munoz, J. A. Bagnell, N. Vandapel, and M. Hebert, "Contextual classification with functional max-margin Markov networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Miami Beach, FL, USA, Jun. 2009, pp. 975–982.

[44] S. Qiu, S. Anwar, and N. Barnes, "Semantic segmentation for real point cloud scenes via bilateral augmentation and adaptive fusion," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 1757–1767.
[45] A. Burt, M. Disney, and K. Calders, "Extracting individual trees from LiDAR point clouds using treeseg," *Methods Ecol. Evol.*, vol. 10, no. 3, pp. 438–445, Dec. 2018, doi: 10.1111/2041-210x.13121.
[46] R. Ferrara, S. G. P. Virdis, A. Ventura, T. Ghisu, P. Duce, and G. Pellizzaro, "An automated approach for wood-leaf separation from terrestrial LiDAR point clouds using the density based clustering algorithm DBSCAN," *Agricult. Forest Meteorol.*, vol. 262, pp. 434–444, Nov. 2018, doi: 10.1016/j.agrformet.2018.04.008.

JUAN LEI received the B.Sc. degree from the School of Surveying and Land Information Engineering, Henan Polytechnic University, Jiaozuo, China, in 2016. She is currently pursuing the master's degree with the School of Geo-Science and Technology, Zhengzhou University, Zhengzhou, China. Her research interests include deep learning-based semantic segmentation algorithm research, 3D point clouds-based automated extraction of urban components, and their applications.

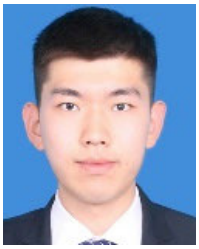
HONGWEI LI is currently a Professor at the School of Geo-Science and Technology, Zhengzhou University, Zhengzhou, China. His research interests include geospatial data mining, spatial ontology and semantic computing, machine vision measurement, and maps and spatial cognition.



SHAN ZHAO is currently a Lecturer at the School of Geo-Science and Technology, Zhengzhou University, Zhengzhou, China. Her research interest includes geographic information visualization and its applications.

YONGJI WANG received the Ph.D. degree in cartography and geographical information system from the State Key Laboratory of Resources and Environmental Information System, Institute of Geographical Sciences and Natural Resources Research, Chinese Academy of Sciences, China, in 2020. He is currently a Lecturer with the School of Geoscience and Technology, Zhengzhou University. His research interests include remote sensing image processing and GIS spatial analysis.

YIRUI JIANG is currently pursuing the Ph.D. degree with the School of Computer and Artificial Intelligence, Zhengzhou University, Zhengzhou, China. Her research interests include smart city management and transit optimization.



GE ZHU received the B.Sc. degree from the School of Surveying and Land Information Engineering, Henan Polytechnic University, Jiaozuo, China, in 2016. He is currently pursuing the master's degree with the School of Geo-Science and Technology, Zhengzhou University, Zhengzhou, China. His research interests include fusion technology of 3D GIS and video data.