A Two-Stage Approach for Individual Tree Segmentation From TLS Point Clouds

Lihong Chang, Hongchao Fan, Ningning Zhu, and Zhen Dong, *Member, IEEE*

***Abstract*—Individual tree segmentation in forest scenes provides a foundation for forest ecosystem modeling and biodiversity assessment applications. Existing approaches work well for cases where trees do not grow in layers. However, they may fail in the scenario with understory vegetation occlusion and heavily overlapped crowns. In this article, we propose a two-stage solution for individual tree segmentation. This method combines a semantic segmentation module and an instance segmentation module. In the first stage, the semantic segmentation network classifies the point clouds into tree and nontree points. In the second stage, the instance segmentation module is utilized by incorporating object detection and postprocessing refinement. The combination of semantic network and object detection network roughly extracts trees, filters out the understory vegetation that affects the extraction of small trees, and improves the extraction probability of small trees. Meanwhile, the method of object detection extraction trees can solve tree extraction omissions due to unclear stems. For the overlap crown, first, object detection limits the border of the tree crown of an individual tree, and further segmentation was implemented by refining clustering. Experiments show that our method solves the above-mentioned problems and achieves state-of-the-art completeness and mean accuracy performances on benchmark datasets.**

***Index Terms*—Individual tree segmentation, instance segmentation, semantic segmentation, terrestrial laser scanning (TLS).**

I. Introduction

PRECISE measurement of forest resources is beneficial for effectively managing resources and sustainable development of forests [5]. As the basic unit of forest resources, the attributes of individual trees like position, tree species, diameter at breast height (DBH), and tree height are the basis of quantifiable forest analysis, biodiversity assessment, and forest ecosystem modeling [2], [35]. During the last few years, LiDAR has provided an efficient method for forest resource inventory [1], [41]. Airborne LiDAR systems (ALS) are widely applied to estimate forest carbon stock and forest biomass in large-scale forest scenes [4], [8]. However, owing to the low point density and occlusion caused by the tree crown, it is not easy to precisely acquire the morphological parameters of individual trees. Meanwhile, the terrain LiDAR system collects point clouds from a side view and close distance [9]. This method can provide high-density point clouds of the lower canopy (e.g., tree trunks and leaves), more suitable for individual tree morphological parameter calculations [28]. Segmenting the individual tree based on the TLS data is critical for acquiring individual tree information [30]. Existing approaches to segmenting individual trees can be categorized into traditional and deep learning-based techniques.

Traditional individual tree segmentation from TLS point clouds generally contains two steps: terrain filtering and single tree isolation [26]. Terrain filtering separates point clouds into the ground and non-ground points [36] and has achieved promising results so far [13], [18]. After removing the ground points, the process of individual tree segmentation contains two subtasks: detection of tree stems and crown segmentation. For the first part, many researchers have worked on it. For example, some researchers have used circle or cylinder fitting algorithms to retrieve the stem [24], [27]. Although such stem detection algorithms have achieved good performance, they suffer from problems in natural forests with high understory vegetation and stem density [11]. Therefore, some researchers have considered directly classifying point clouds to extract stem or wood points based on these challenges. These approaches can be classified into segment-based and point-based. Point-based methods utilize the geometric features of 3-D coordinates based on facts to acquire the stem or wood points and achieve satisfactory classification accuracy [38]. However, the classification strategy requires setting prior parameters, and unreasonable parameters can affect the classification accuracy of the point clouds. The segment-based approaches first grouped the point clouds into segments and then classified the segments by the feature (e.g., the size or shape of a segment). Compared with point-based classification, segment-based methods reduce computing resources. However, these methods are widely used in urban environments [25] and require further exploration in forest scenes.

Manuscript received 15 May 2022; revised 12 July 2022 and 14 August 2022; accepted 20 September 2022. Date of publication 6 October 2022; date of current version 17 October 2022. This work was supported by the National Natural Science Foundation of China under Grant 41771484. *(Corresponding author: Zhen Dong.)*

Lihong Chang is with the School of Remote Sensing and Information Engineering, Wuhan University, Wuhan 430079, China (e-mail: 2017286190147@whu.edu.cn).

Hongchao Fan is with the School of Remote Sensing and Information Engineering, Wuhan University, Wuhan, China, and also with the Department of Civil and Environmental Engineering Faculty of Engineering, Norwegian University of Science and Technology, 7491 Trondheim, Norway (e-mail: hongchao.fan@ntnu.no).

Ningning Zhu is with the State Key Laboratory of Information Engineering in Surveying, Mapping and Remote Sensing, Wuhan University, Wuhan 430079, China (e-mail: ningningzhu@whu.edu.cn).

Zhen Dong is with the School of Remote Sensing and Information Engineering, Wuhan University, Wuhan, China, and also with the State Key Laboratory of Information Engineering in Surveying, Mapping and Remote Sensing, Wuhan University, Wuhan 430079, China (e-mail: dongzhenwhu@whu.edu.cn).

Digital Object Identifier 10.1109/JSTARS.2022.3212445

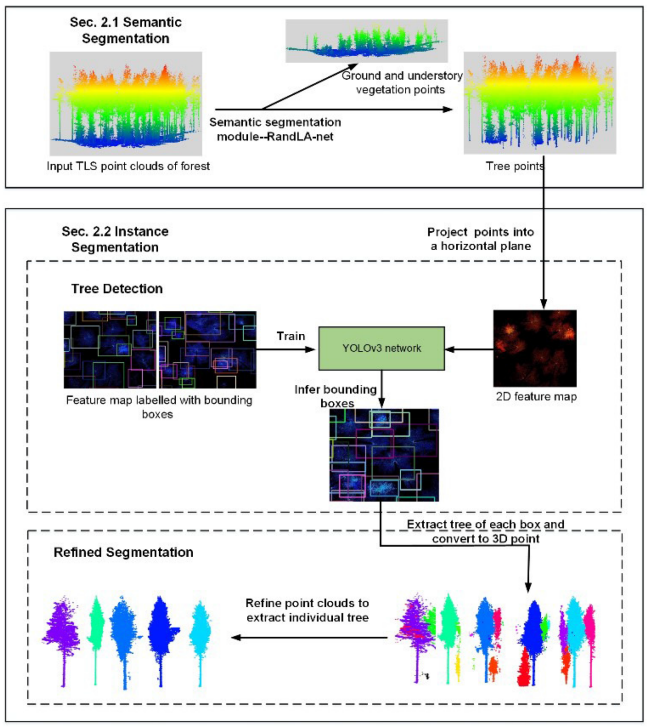
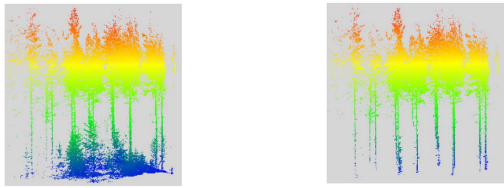


Fig. 1. Proposed architecture for individual tree segmentation.



(a) (b)

Fig. 2. Illustration of semantic segmentation for nontree points removal. (a) Original point cloud. (b) Tree points after removing the nontree points.

Compared to the detection of tree stems, crown segmentation has received less attention. Ren et al. [22] used the statistical characteristics of point clouds along the vertical direction to extract trees from ground vegetation. Guan et al. [14] segmented trees from mobile laser scanner data utilizing Euclidean distance clustering and voxel-based normalized cuts. Yadav and Lohani [33] identified tree trunks first. Then they retrieved the crown according to its nearly circular feature on cross-section and symmetry based on the stem. These methods are mainly used in urban environments with regularly arranged trees and little understory vegetation, which encounter difficulties in forest scenes. Several algorithms have been proposed to deal with the problem of tree crown segmentation in forests. Olofsson et al. [19] detected the stem and then utilized the symmetric image to detect the tree crown. Raumonen et al. [20] segment trees based on morphological rules. The stems were detected first, and trees were grown step-by-step through the connectivity assumption and orientation. These methods perform effectively for a sparse and straightforward forest with a wide area and lesser overlap between canopies. However, they are challenging to transfer to more complex forest types. Several attempts have been made to deal with the crown segmentation of difficult forest types. Yang et al. [34] first extracted the individual stem, then utilized a hierarchical minimum cut method to segment the crown. The results of crown segmentation were affected by the order of segmentation. Burt et al. [3] extracted stems first and then segmented tree crowns by region-based segmentation and connectivity testing. The shortcoming of this method is that it occasionally demands the removal of neighboring vegetation manually. Yun and Zheng [39] segmented the overlapped tree crown based on the seed points identification and DBSCAN algorithm, and the parameters have many effects on the final results. Itakura [40] combined tree detection and watershed algorithm to segment trees. This method ignored the vertical forest structure and caused problems with crown segmentation.

Recently, deep learning-based approaches have developed rapidly and are broadly used in various fields. Because of the outstanding potential of feature extraction in point clouds, many researchers have attempted to apply it to forest scenes, which mainly include classification and tree extraction. For instance, Xi and Hopkinson [32] utilized a deep-learning approach to filter wood points and classify tree species from TLS data. These classification algorithms have proven effective in wood-leaf classification, and species classification and effectiveness can be further explored for different categories. For tree extraction, some attempts also have been made. Wang et al. [28] applied faster R-CNN [to extract stems from point clouds. Chen et al. [7] identified tree crown points based on the PointNet [6] deep learning framework at the voxel scale. This method requires a voxel of the point clouds first, which leads to a reduction in accuracy and time-increasing. Some scholars extracted trees based on images. For example, Windrim and Bryson [29] applied faster R-CNN [22] to detect trees from 2-D images based on ALS datasets. Xi et al. [31] adopted the CenterNet [10] model to detect crowns from densely overlapped crowns based on TLS datasets. The detected crown regions were used as a fundamental reference for the crown-based attribute analysis. Although these methods achieved good results on crown detection based on 2-D images, tree segmentation on 3-D points has been ignored.

In conclusion, the existing methods achieved good performance on individual tree segmentation in the straightforward plot. However, they suffer from the quality of segmentation in complex plots, and there remains considerable room for improvement. For instance, Long et al. [17] achieved 30% recall on tree detection in a single-scan plot, Burt et al. [3] segment crown, about 30% required further manual segmentation inside a tropical forest plot. These reveal in the arguments as follows. First, the terrain filtering methods only remove the ground points, whereas the remaining understory vegetation seriously after ground point filtering deteriorates the performance of single tree locating and tree extractions, especially for areas with understory vegetation attached to the tree stems. Second, the existing methods focus on single tree locating by tree stem extraction. At the same time, crown segmentation receives less attention and achieves non-ideal performance for heavily overlapped crowns, thus resulting in inaccurate morphological parameters (e.g., crown diameter and tree height) calculation. Third, most of the previous methods concentrate on evaluating

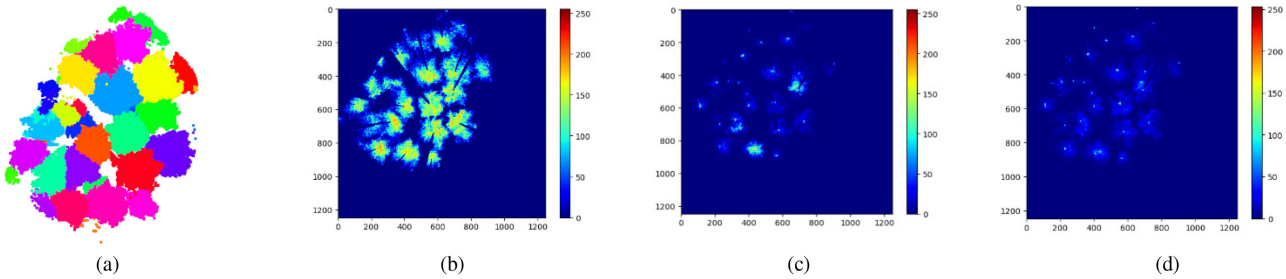


Fig. 3. Point distribution from a bird's eye view. (a) Top view distribution. (b) Maximum height distribution. (c) Height difference distribution. (d) Point density distribution.

the tree location results, and the quantification of pointwise individual tree segmentation has not yet been implemented. To overcome the limitations mentioned above, we propose a two-stage individual tree segmentation approach that combines the semantic segmentation network for the discrimination of tree and nontree points and the instance segmentation network for accurate individual tree segmentation. The main contributions of this study are as follows.

1) A novel two-stage individual tree segmentation approach is proposed by combining a semantic segmentation module and an instance segmentation module.
2) The combination of a semantic segmentation network and object detection network to extract trees not only resolve the extraction problem of small trees but also solve tree extraction omissions due to unclear stems.
3) The combination of an object detection network and refined segmentation provides a solution for the segmentation of overlapping tree crowns.

This article is organized as follows. Section II thoroughly described the proposed approach. Then, Section III presented the experimental results. After that, Section IV is a discussion of the approach. Finally, Section V concludes this article.

II. Two-Stage Approach

The overall framework of the proposed approach is shown in Fig. 1. The approach contains two main modules: a semantic segmentation module and an instance segmentation module. The first is the semantic segmentation module, which effectively filters out the ground and low vegetation points (see Section II-A). The second module defines the instance segmentation module to detect individual trees from the remaining points and refine them using a series of clustering operators (see Section II-B).

A. Semantic Segmentation for Nontree Points Removal

The general terrain filtering methods filter the ground points only, and the left understory vegetation points heavily affect the performance of single tree locating. Therefore, we proposed the semantic segmentation network to classify the point clouds into the tree and nontree points to overcome this limitation. RandLA-net [12] was deployed in this process because of its high accuracy and efficiency for large-scale point cloud processing. The network architecture of RandLA-net is modified based on the following aspects. First, because it classifies tree and nontree points, the prediction classification is set as two. Second, the negative log-likelihood loss function is applied to the training model for unbalanced samples. The weights of different types are determined by the inverse proportion of the data size. Third, many overlapped samples were collected to reflect the data distribution fully. Therefore, the final predicted result of each point was determined by voting on several predicted results, and the number of votes was set as five.

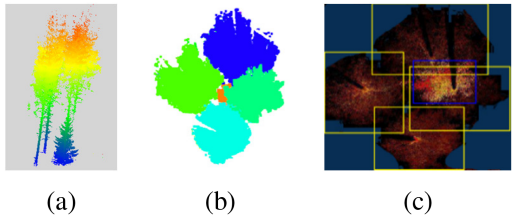


Fig. 4. Result of individual tree detection. (a) Side view of trees containing connected crowns and small trees. (b) Top view of trees. (c) Detection result.

After this process, the rest points are provided for further individual tree segmentation. As shown in Fig. 2, removing the non-tree points decreases the computational burden of subsequent segmentation and makes tree distribution more obvious.

B. Individual Tree Segmentation

In this process, the individual tree segmentation is composed of three key steps: feature map generation, individual tree detection, and postprocessing refinement. Object detection preliminarily defines the tree contour, which is further optimized in refined segmentation.

1) Feature Maps Generation: According to ecological and botanical rules, tree crowns are theoretically close to their respective stems/roots in geographic space. When observing the points of trees from a bird's perspective, the point clouds of each tree are gathered around its own highest point, and clusters segment the entire forest. We projected the point clouds into a horizontal plane in the bird's eye view to generate feature maps based on these features. More specifically, we use reasonable features for elaborate feature map generation based on the following observations.

1) As shown in Fig. 3(b), each tree is centered and clustered around the local peak, and thus, the highest point of each grid was introduced into the feature map. Specifically, the

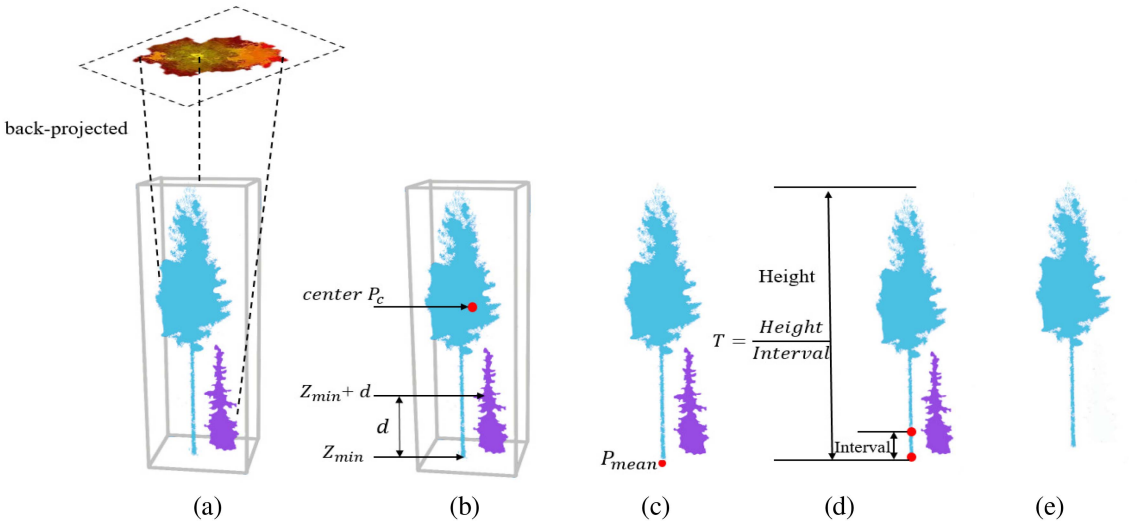


Fig. 5. Key steps of refining segmentation. (a) Process of back projection. (b) Result of some parameters calculation. (c) Result of P_{mean}, from the first step in refined segmentation. (d) Parameter T. (e) Result of refined segmentation from the second step.

maximum z value in each grid was set as the maximum height feature.

2) As shown in Fig. 3(c), the height difference of the tree radiates outward from the highest point and gradually decreases. Therefore, we adopted the height difference as a feature in the feature map. Specifically, the difference between the maximum z value and the minimum z value of each grid was set as the height difference feature.

3) As shown in Fig. 3(d), the point density reflects the degree of point aggregation, and the location of the tree stem has the highest point density. Therefore, the relative point density is introduced into the feature map. Specifically, the relative point density feature was calculated as the ratio between the number of points and the maximum number of points in each grid.

Finally, the features of maximum height, height difference, and relative point density are normalized to 0–255 and set as the feature maps' red, green, and blue channels, respectively.

2) Individual Tree Detection: Individual trees were detected from the feature maps using the YOLOv3 network [21], and other object detection networks (e.g., [2] and [17]) also can be used. YOLOv3 detects objects by learning global information in the image, mainly including the prediction of the bounding box, confidence, and class score. The input feature map is divided into $M\times M$ grid cells, and the predicted bounding box is represented by (x, y, w, h). When the method is applied to the forest scene, YOLOv3 predicts a 3-D tensor for each scale of output: $M\times M\times [3*(4+1+1)]$, which represents four parameter values of prediction, that is, the scales, the coordinates of the bounding box, the confidence of bounding box, and the category. After individual tree detection, each bounding box contains a complete tree even though on connected crowns, and the small trees in the lower canopy can also be detected (see Fig. 4).

3) Refined Segmentation: The individual detected tree from the individual tree detection stage limits the individual tree and tree crown boundary and simplifies the subsequent calculation.

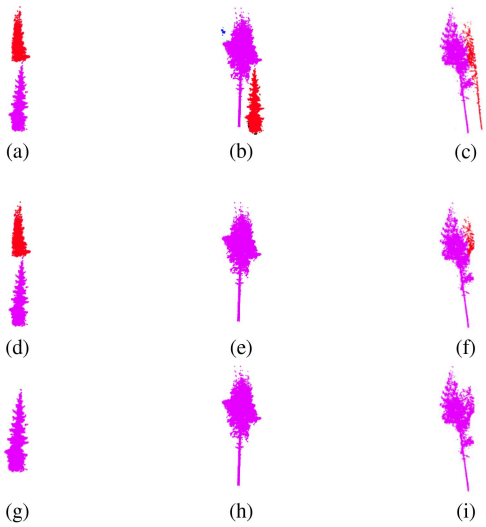


Fig. 6. Process of refining segmentation. (a), (b), and (c) First, second, and third category problems, respectively, when back-projecting 2-D images to 3-D point clouds. (d), (e), and (f) Results of primary tree extraction based on hierarchical clustering from first, second, and third category problems, respectively. (g), (h), and (i) Result of segmentation from first, second, and third category problem, respectively.

In this process, the pixels are back-projected to point clouds according to the correspondence between the pixels and the point clouds [see Fig. 5(a)], and then each point is assigned a unique instance code. Each bounding box in the feature map is a detected tree. Theoretically, the points within this bounding box belong to a detected tree. Unfortunately, the image can reflect the 2-D planar features but cannot describe the vertical features. Hence, there are some problems when back-projecting 2-D images to 3-D point clouds to obtain an individual tree. These problems can be divided into three categories. For the first category [see Fig. 6(a)], one bounding box contains the points of a small tree and parts of points from another giant tree above this small

tree. In the second category (see Fig. 6(b)), one bounding box detected a large tree, but the small tree under this large tree was also detected. The third category occurred in areas with heavily overlapped crowns and high-density trees. Fig. 6(c) shows that one bounding box contains the main tree and part of another tree. To solve these problems, a segmentation refinement procedure was proposed.

The center point $P_i(x_i, y_i, z_i)$ of the main tree is determined in the first step. Calculate the center coordinate $P_c(x_c, y_c, z_c)$ and minimum z value $Z_{\min}$ of all points, select the points C where the z value ranges from $Z_{\min}$ to $Z_{\min} + d$, which means the distance [see Fig. 5(b)]. Then, point clouds C are grouped into clusters using the hierarchical clustering algorithm. Select the center coordinate $P_{\text{mean}}(x_{\text{mean}}, y_{\text{mean}})$ [see Fig. 5(c)] of the applicable cluster close to the center P_c on the *xoy*-plane as the center of the main tree.

In the second step, hierarchical Euclidean clustering and Hierarchical Clustering were used to extract the main tree. Point clouds V in the bounding box are segmented into a set of point clouds $\{R\}$ along the vertical direction according to the count of segmentation,T [see Fig. 5(d)]. Commencing from P_{mean} on the *xoy*-plane, each R_i, $R_i \in \{R\}$ is segmented by the Euclidean clustering and hierarchical clustering algorithms. Fig. 6(d)–(f) shows the results.

After these two steps, there are still some problems. For example, the first category problem still exists. Therefore, a mean-shift clustering algorithm was utilized to solve the first category problem. The final results are shown in Fig. 6(g)–(i).

The effectiveness of clustering depends on the detection process. The combination of detection and clustering together leads to good segmentation.

III. Experiment

We implemented the proposed approach and tested it using an international TLS benchmarking project [16] and Qintang Forest.

A. Experimental Data

This study utilized the TLS point clouds dataset from the international TLS benchmarking project [16]. The benchmark datasets were collected from a southern boreal forest in Evo, Finland (61.19 °N, 25.11 °E). Based on the difference in forest-stand conditions, they collected 24 sample plots. Varying understory vegetation and stem densities classified these plots into three categories: easy, medium, and difficult. The "Easy" represented that few understory vegetation and the stem is visible with lower density. The "Medium" plot possesses moderate stem density, and understory vegetation is sparse. The "Difficult" has high stem density and understory vegetation. The main tree species in these sample plots were Scots pine, Norway spruce, silver, and downy. The reference data were gathered through field surveys and manual measurements. The researchers provided six sample plots containing field reference data for nonprofit research. Both single- and multiscan policies are adopted for data collection. We manually labeled 26 sample plots in a pointwise manner with our point cloud annotations tool developed based on CloudCompare (https://www.cloudcompare.org/) by using Qt and OpenGL, and each point in each plot was assigned to a semantic label and an instance label. Then, the object bounding boxes of the feature maps were calculated according to the corresponding instance annotation.

TABLE I
Detailed Description of Annotation Data

Number of nontree points, tree points, trees					
Training samples			535,681,772	386,220,141	1127
Testing samples	**Multi scan**	Plot1 (**Easy**)	71,668,887	39,396,306	57
		Plot2 (**Easy**)	71,761,411	41,962,680	88
		Plot3 (**Medium**)	61,953,716	58,003,675	123
		Plot4 (**Medium**)	70,952,637	58,198,685	63
		Plot5 (**Difficult**)	63,445,088	61,062,080	99
		Plot6 (**Difficult**)	67,784,130	43,284,446	153
		Plot7 (**--**)	19,674,321	13,659,961	43
	Single scan	Plot1 (**Easy**)	13,911,871	9,418,699	48
		Plot2 (**Easy**)	14,417,500	9,005,962	78
		Plot3 (**Medium**)	11,608,616	12,029,172	104
		Plot4 (**Medium**)	12,974,554	14,157,172	55
		Plot5 (**Difficult**)	12,557,265	12,909,956	60
		Plot6 (**Difficult**)	12,388,585	10,084,917	78

To verify the applicability of the designed method, we introduced and labeled another sample plot located in the Qintang Forest in Guigang City, Guangxi Zhuang Autonomous Region, China, hereafter referred to as plot 7. We set up five ground scanning stations in this area, with a scanning resolution of 0.03 °. We only selected the point clouds within 40 m from the ground station because of the occlusion. The sample plot was mainly composed of pine trees and included a small number of buildings, wire poles, and other artificial structures.

This experiment used six public sample plots with reference data and Plot 7 as testing samples and other plots as training samples. Table I provide a detailed explanation of annotation data. Fig. 7 shows part of the results of the annotation. The blue and green represent the tree points and nontree points, respectively.

B. Implementation Details

For RandLA-net, we used the SGD as an optimizer to train the network. The grid size for the downsample is 5 cm. The method Xavier used to initialize the weight. The initial learning rate was set to 0.01, and the learning rate decreased by 0.05 every 10 epochs. The size of the input point clouds for each block was 40960. We sampled 2760 batches from the sample plots. The epoch number is 100, and the batch size is 6.

The YOLOv3 network uses DarkNet-53 for training and testing. Before training, we used the convolution weights of the pre-trained model to perform transfer learning. The size of

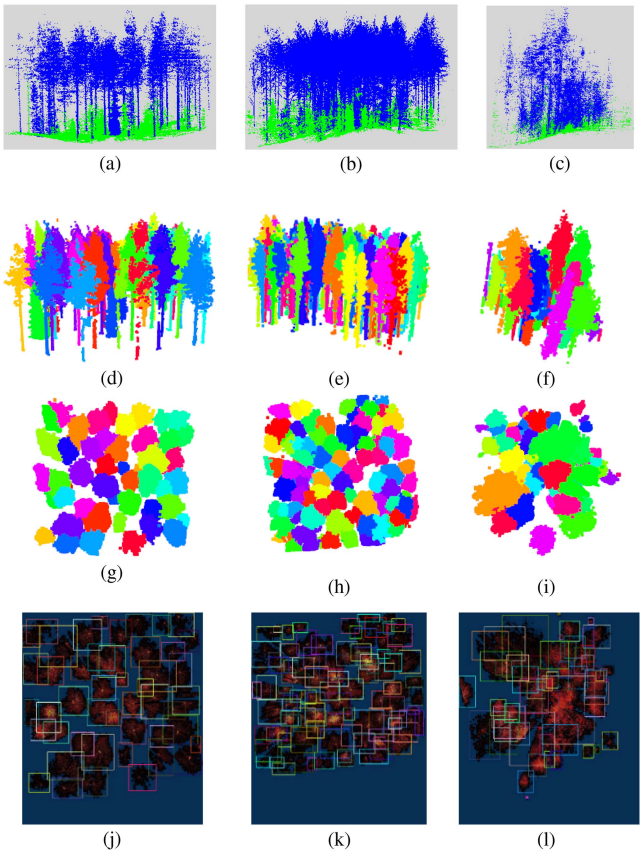


Fig. 7. Data annotated in various forms. (a), (b), and (c) Results of semantic annotation from easy, medium, and difficult sample plots, respectively. (d), (e), and (f) Results of instance annotation from easy, medium, and difficult sample plots, respectively. (g), (h), and (i) Top view results of instance annotation from easy, medium, and difficult sample plots, respectively. (j), (k), and (l) Results of bounding box annotation from easy, medium, and difficult sample plots, respectively. (While the green points in (a), (b), and (c) present nontree points, the blue points represent trees.)

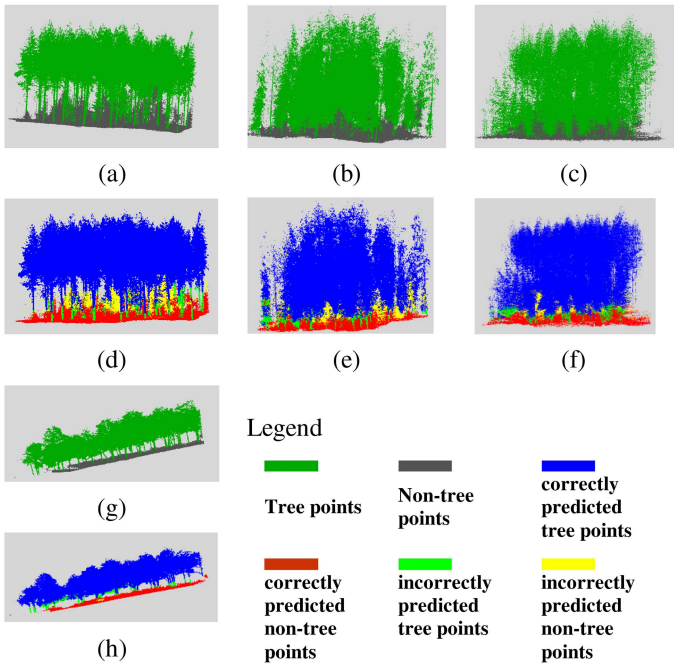


Fig. 8. Semantic segmentation results. (a), (b), (c), and (g) Original annotation from the easy plot, medium plot, difficult plot, and Plot 7, respectively. (d), (e), (f), and (h) Classification results from easy plot, medium plot, difficult plot, and Plot 7, respectively.

the input image was 416 × 416 pixels. We used SGD as the optimizer; 0.001 was the value of the initial learning rate. The value of momentum was 0.9. The 2372 batches were used as training, and the batch size is 8.

Python3.8 was used to train and test the datasets. Two GeForce GTX TITAN X were used in the experiments.

In refining individual trees, the distance d is set as 4 and T of the hierarchical Euclidean and hierarchical clustering was set as 15.

C. Experimental Results

We presented and evaluated the results from four aspects: semantic segmentation, individual tree detection, single tree locating, and parameter calculation.

1) Semantic Segmentation: Because the primary purpose of semantic segmentation is to extract tree points, we mainly evaluated the accuracy of tree point extraction. The quantitative evaluation metrics for tree point classification are precision, recall, F1-score, and overall accuracy (OA). TP, FN, FP, and TN indicate the number of true positives, false positives, false negatives, and true negatives. The process of calculation can be formulated as follows:

$$\text{Precision} = 100\% \times \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{1}$$

$$\text{recall} = 100\% \times \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{2}$$

$$F1 - \text{score} = \frac{2 \times \text{Precision} \times \text{recall}}{(\text{precision} + \text{recall})} \tag{3}$$

$$\text{OA} = 100\% \times \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FN} + \text{FP} + \text{TN}}. \tag{4}$$

The semantic segmentation results in the "Easy" plot, "Medium" plot, "Difficult" plot, and Plot 7 are shown in Fig. 8(a)–(c) and (g), respectively, where green and gray represent tree points and nontree points. Fig. 8(d)–(f) and (h) represent four types of points: correctly predicted tree points in blue, incorrectly predicted tree points in green, correctly predicted nontree points in red, and incorrectly predicted nontree points in yellow. Visualization results showed that most points were correctly classified. The specific accuracy of semantic segmentation is shown in Table II.

The recalls of all sample plots are more significant than 0.9, which means that most tree points are correctly obtained under varying forest-stand conditions. The purpose of semantic segmentation is to filter out the ground and low vegetation points and detect trees as much as possible. Thus, tree points are expected to be preserved as much as possible, and nontree points predicted as tree points are hoped to have a mini impact

TABLE II
SEMANTIC SEGMENTATION RESULTS

Scan mode	Plot	*OA*(%)	*Precision* (%)	*Recall* (%)	*F-score* (%)
Multi scan	**Plot1**	94.2	95.0	97.1	96.1
	Plot2	95.6	95.8	98.3	97.0
	Plot3	91.5	93.1	96.2	94.7
	Plot4	91.9	91.7	98.2	94.8
	Plot5	93.4	95.3	96.1	95.8
	Plot6	87.6	88.7	95.3	91.9
	Plot 7	94.8	97.8	95.7	96.7
	Mean	**92.7**	**93.9**	**96.7**	**95.3**
Single scan	**Plot1**	94.5	96.9	95.2	96.1
	Plot2	94.5	94.5	97.5	96.0
	Plot3	91.3	95.1	92.3	94.0
	Plot4	91.6	91.8	96.6	94.2
	Plot5	87.1	90.7	91.3	91.0
	Plot6	76.3	73.8	92.4	82.1
	Mean	**89.2**	**90.5**	**94.3**	**92.2**

on tree distribution. Therefore, high recall guarantees the correct distribution of trees.

2) Individual Tree Detection: The results of tree detection were assessed on feature maps. The general evaluation factor, intersection over union (IOU), was applied as the basis of the assessment metrics. If the IOU is greater than 0.5, the tree is correctly detected. The specific quantitative evaluation metrics are precision (p), recall (r), and F1-score ($F1$). TP, FN, FP, and TN indicate the number of true positives, false positives, false negatives, and true negatives. The process of calculation can be formulated as follows:

$$p = 100\% \times \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}} \tag{5}$$

$$r = 100\% \times \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}} \tag{6}$$

$$F1 = \frac{2 \times p \times r}{(p + r)}. \tag{7}$$

The specific accuracy of tree detection is shown in Table III. The recall of all sample plots was more significant than 0.9, which means that most of the annotation trees were detected correctly. While the precision of tree detection is less than 0.9. Recall and precision have a relationship. We want to preserve the tree as much as possible, so we lose some precision. The precision of multiple scan data is better than single scan data.

3) Single Tree Locating: In this process, we assumed that the stem was approximately cylindrical. The DBH and position of each tree were measured from the diameter and horizontal coordinates of the fitted cylinder 1.2–1.4 m above the ground. However, there are existing some problems. For example, the absence of data makes the DBH and horizontal coordinates impossible to obtain. Therefore, the average at other heights would be calculated and considered as the DBH and position. If the DBH and horizontal center coordinates could not be obtained owing to the cylinder fitting failure, the averaged horizontal coordinates and vertical difference were calculated as the position and height of the tree. If the tree's position and height matched the reference tree's parameter, the tree was considered to be correctly located. Stems with a DBH greater than 5 cm were classified as detected trees.

The results of individual tree locating were evaluated by the indicators from the TLS benchmark project. n_{match}, n_{ref}, and n_{extr} represent the number of correctly detected trees, reference trees, and detected trees, respectively. The specific definitions are given in the following equations:

$$\mathrm{Completeness} = \frac{n_{\mathrm{match}}}{n_{\mathrm{ref}}} \tag{8}$$

$$\mathrm{Correctness} = \frac{n_{\mathrm{match}}}{n_{\mathrm{extr}}} \tag{9}$$

$$\mathrm{Mean\ accuracy} = \frac{2n_{\mathrm{match}}}{(n_{\mathrm{ref}} + n_{\mathrm{extr}})}. \tag{10}$$

Table IV shows the results. Compared to the single-scan TLS datasets, the multiscan TLS datasets achieved better performance on the evaluation factor. The completeness and mean accuracy of locating a single tree decreased with forest stand complexity increasing. The DBH was underestimated in all sample plots. The mean bias of DBH was 2.82 and 3.3 cm for the multiscan and single-scan datasets, respectively. The estimation results of DBH were comparable to those from [16].

4) Parameters Calculation: Almost all previous methods evaluated tree location only, and the evaluation of pointwise individual tree segmentation has not been applied. Therefore, the results of individual tree segmentation at the point level were evaluated, and the IOU was utilized as the basis of evaluation metrics. TP, FN, and FP indicate the points number of true positives, false positives, and false negatives, respectively. IoU_i represents the IOU that every correctly detected tree i. The mean IOU is represented by mIoU. *N* represents the number of correctly detected trees. The specific definitions are given in the following equations:

$$\mathrm{IoU}_i = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP} + \mathrm{FN}} \tag{11}$$

$$\mathrm{mIoU} = \frac{\sum_{i=1}^{N} \mathrm{IoU}_i}{N}. \tag{12}$$

The structural parameters are the primary goal of individual tree segmentation. Moreover, estimating tree height and crown diameter is the main challenge. Thus, we evaluated the accuracy of the crown diameter and tree height. *K* is the number of observation data, $\widehat{y_i}$ denotes the reference value, and $\bar{y}$ is the variable's mean. The specific definitions are given by the following equation:

$$\mathrm{RMSE} = \left(\frac{1}{k} \sum_{i=1}^{k} (y_i - \widehat{y_i})^2 \right)^{1/2} \tag{13}$$

$$\mathrm{RMSE}\,(\%) = 100\% \times \frac{\mathrm{RMSE}}{\bar{y}}. \tag{14}$$

In this process, the annotation crown diameters were set as the reference data, and we evaluated the accuracy of crown diameters only on multiscan datasets. The specific accuracy of instance segmentation is shown in Table V.

TABLE III
ACCURACY ASSESSMENT OF INDIVIDUAL TREE DETECTION

Scan mode	Plot	Numbers of trees(annotation)	*TP*	*FP*	*FN*	r(%)	p(%)	$F1$(%)
Multi scan	Plot1	57	57	1	0	100.0	98.3	99.1
	Plot2	88	86	8	2	97.7	91.5	94.5
	Plot3	123	120	16	3	97.6	88.2	92.7
	Plot4	63	63	9	0	100.0	87.5	93.3
	Plot5	99	98	19	1	99.0	83.8	90.7
	Plot6	153	150	23	3	98.0	86.7	92.0
	Plot 7	43	43	3	0	98.9	89.9	94.1
	Overall	**626**	**617**	**79**	**9**	**98.9**	**89.9**	**94.1**
Single scan	Plot1	48	48	1	0	100.0	98.0	99.0
	Plot2	77	73	23	4	94.8	76.0	84.4
	Plot3	104	103	15	1	99.0	87.3	92.8
	Plot4	54	53	9	1	98.1	85.5	91.4
	Plot5	60	58	17	2	96.7	77.3	85.9
	Plot6	72	68	24	4	94.4	73.9	82.9
	Overall	**415**	**403**	**89**	**12**	**97.2**	**83.0**	**89.4**

TABLE IV
ACCURACY OF SINGLE TREE LOCATING

Scan mode	Plot Complexity	Stem detection			Location	DBH	
		Completeness (%)	Correctness (%)	Mean accuracy (%)	RMSE (m)	RMSE (cm)	Bias (cm)
Multi scan	Plot1	94.1	94.1	94.1	0.10	1.01	−0.3
	Plot2	94.0	89.7	91.9	0.18	2.59	−0.8
	Plot3	78.4	94.3	85.6	0.19	2.89	−0.7
	Plot4	74.3	93.5	82.8	0.15	2.69	−1.4
	Plot5	64.8	85.8	73.9	0.14	4.10	−1.38
	Plot6	52.7	81.6	64.1	0.15	3.66	−1.57
	Plot 7	100.0	97.7	98.8	–	–	–
	Mean	**79.7**	**90.9**	**84.5**	**0.15**	**2.82**	**−1.03**
Single scan	Plot1	88.2	95.7	90.9	0.17	2.82	−1.1
	Plot2	63.1	94.6	75.7	0.12	2.76	−1.0
	Plot3	52.0	95.1	68.4	0.14	2.9	−1.3
	Plot4	44.9	94.6	60.9	0.16	3.3	−1.4
	Plot5	22.9	61.0	33.3	0.18	4.5	−2.2
	Plot6	12.0	50.7	19.0	0.17	3.6	−1.4
	Mean	**47.2**	**82.0**	**58.0**	**0.16**	**3.3**	**−1.4**

TABLE V
RESULTS OF INSTANCE SEGMENTATION AND TREE PARAMETERS EVALUATION

Plot complexity	mIoU(%)		Crown diameter		Tree height			
	Multi scan	Single scan	RMSE (m)	RMSE (%)	Multi scan		Single scan	
					RMSE (m)	RMSE (%)	RMSE (m)	RMSE (%)
Plot1	90.1	85.3	0.21	5.66%	1.02	5.3%	2.06	10.60%
Plot2	85.0	74.5	0.34	12.7%	1.51	14.3%	2.48	17.70%
Plot3	82.1	76.6	0.42	14.8%	1.69	10.1%	3.49	18.20%
Plot4	80.3	76.7	0.45	20.0%	2.37	12.3%	6.8	34.50%
Plot5	79.8	69.6	0.68	23.3%	2.73	18.8%	5.67	36.90%
Plot6	73.1	63.8	0.56	19.8%	3.35	22.7%	5.26	33.9%
Plot 7	84.5	–	0.82	9.3%	–	–	–	–
Mean	**82.1**	**74.4**	**0.50**	**15.1%**	**2.11**	**13.92%**	**4.29**	**23.58%**

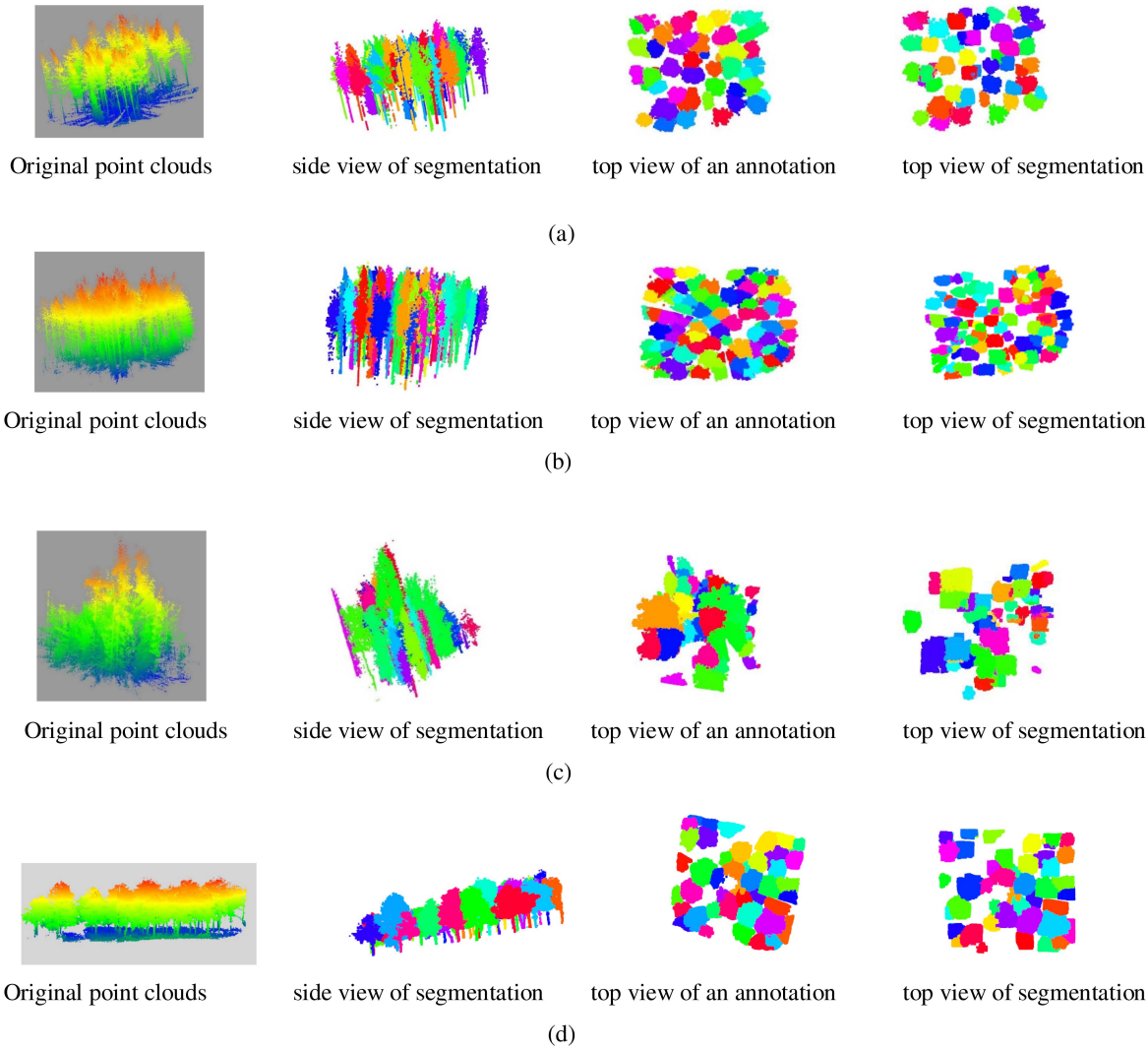


Fig. 9. Individual tree segmentation results on testing plots. (a), (b), (c), and (d) Visualization results from the easy plot, medium plot, difficult plot, and Plot 7, respectively.

The results demonstrated that the accuracy of individual tree segmentation on multiscan TLS datasets was superior to single-scan TLS datasets from mIOU indicators. The average mIoU was 0.821 and 0.744 for the multiscan and single-scan datasets, respectively, implying that the trees were generally well segmented. With forest stand complexity increasing, the mIOU decreased, whereas the RMSE increased. The RMSE and RMSE% of the tree height assessment were approximately 2.11 m, 13.92%, and 4.29 m, 23.58% for the single-scan and multiscan datasets, respectively, comparable with TLS benchmarking project [16]. The visualization results of individual tree segmentation are shown in Fig. 9.

IV. Discussion

A. Single Tree Locating

This study proposed a two-stage method for single-tree segmentation from TLS data and achieved comparable results in locating single trees. Our approach achieved state-of-the-art performance regarding mean accuracy, whether multiscan or single scan (see Tables VI and VII). Meanwhile, the accuracy of DBH is comparable to [16]. There are two main reasons for this result. First, semantic segmentation filtered out the low vegetation points, making the distribution features of trees on the feature map more obvious while reducing the probability of low vegetation being mistaken for trees. Second, we detected the trees from feature maps, which can detect trees with unclear stems but apparent structures easily missed by the stem detection

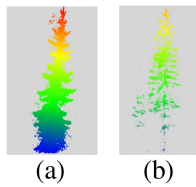


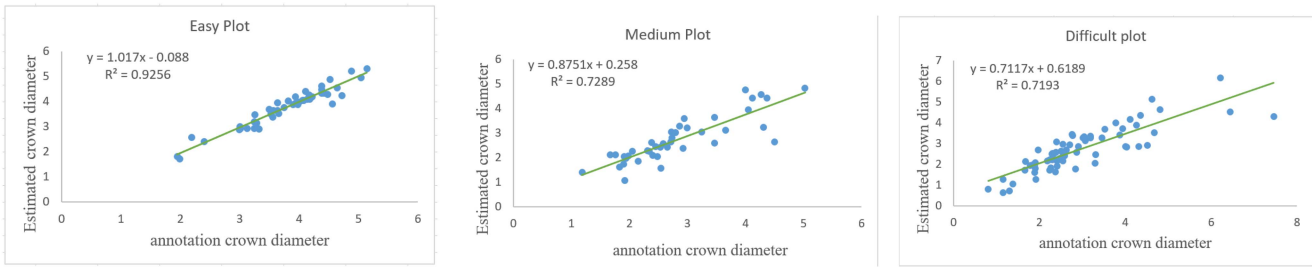
Fig. 10. Trees that are easily missed by the stem detection method.

TABLE VI
COMPARISONS OF SINGLE TREE LOCATING BASED ON MULTISCAN TLS BENCHMARK

Plot complexity	Completeness (%)			Correctness (%)			Mean Accuracy (%)		
	Zhang et al. [36]	Wang [26]	Ours	Zhang et al. [36]	Wang [26]	Ours	Zhang et al. [36]	Wang [26]	Ours
Plot1	86.3	90.2	94.1	97.8	95.8	94.1	91.7	92.9	94.1
Plot2	82.1	92.9	94.0	95.8	81.3	89.7	88.5	86.7	91.9
Plot3	61.5	79.1	78.4	100.0	72.2	94.3	76.2	75.5	85.6
Plot4	57.7	78.2	74.3	97.8	70.1	93.5	72.6	74.0	82.8
Plot5	45.8	64.1	64.8	93.8	67.2	85.8	61.5	65.3	73.9
Plot6	26.3	52.5	52.7	98.4	70.9	81.6	41.5	60.3	64.1
Mean	**59.9**	**76.2**	**76.4**	**97.3**	**76.3**	**89.9**	**72.0**	**75.8**	**82.1**

TABLE VII
COMPARISONS OF SINGLE TREE LOCATING BASED ON SINGLE SCAN TLS BENCHMARK DATASET

Plot	Completeness (%)		Correctness (%)		Mean accuracy (%)	
	Zhang et al. [36]	Ours	Zhang et al. [36]	Ours	Zhang et al. [36]	Ours
Plot1	80.4	88.2	97.6	95.7	88.2	90.9
Plot2	57.1	63.1	92.3	94.6	70.6	75.7
Plot3	46.6	52.0	100.0	95.1	63.6	68.4
Plot4	34.6	44.9	100.0	94.6	51.4	60.9
Plot5	13.7	22.9	90.0	61.0	23.8	33.3
Plot6	8.5	14.4	100.0	50.7	15.6	22.4
Mean	**40.2**	**52.4**	**96.7**	**81.9**	**52.2**	**62.2**



(a) (b) (c)

Fig. 11. Assessments of crown diameter estimations. (a), (b) and (c) Easy plot, medium plot, and difficult plot, respectively.

method. As shown in Fig. 10, the stems are difficult to extract by techniques such as leaf tree classification and cylindrical fitting because of the problematic diagnosis of features; however, our method can extract a tree based on its morphological characteristics.

B. Crown Segmentation

The RMSE and RMSE% of crown diameter showed that the crown could be segmented well. We further analyze the regression of crown diameter. The results are shown in Fig. 11. R^2 for the linear regression for crown diameter estimation in the three types of plots were 0.925, 0.729, and 0.719, respectively. The slopes were 1.017, 0.875, and 0.712. The underestimated crown diameter was caused by refined segmentation, in which the boundary points were discarded.

C. Applicability to Other Forest Scenes

The proposed approach was carried out on Plot 7 with different forest conditions and point densities. The mean accuracy of 98.8% on single tree locating and 0.845 mIoU on tree pointwise segmentation showed that our method can be used in the forest stand which is different from training data.

V. CONCLUSION

Individual tree segmentation of forest point clouds is a significant and primary step in forest resource inventory. This study presented a two-stage individual tree segmentation approach combining a semantic and an instance segmentation network.

We evaluated the designed approach on the Qintang forest and benchmark datasets that contain diverse forest stand with various species, varying stem densities, and a wealth of understory vegetation. Individual tree segmentation was assessed at a point

level for the first time. Furthermore, we compared the designed approach with the current methods, the higher completeness and mean accuracy for the easy, medium, and difficult plots showed that we attained state-of-the-art results on individual tree segmentation. Meanwhile, the proposed instance segmentation module detects individual trees from the elaborate feature maps. Then, the individual detected trees are back-projected to point clouds and refined using a series of clustering operators. This achieves good performance for crown segmentation and tree height calculation. However, this method is verified mainly in coniferous forests. Therefore, the diversity of forest conditions and TLS data is still restricted. Meanwhile, this method only concentrated on the individual tree segmentation and ignored the separation of wood and leaf. Therefore, further studies will be conducted about the usability of this method on the kinds of forests, and we will consider the separation of wood and leaf.

REFERENCES

[1] I. Barbeito et al., "Terrestrial laser scanning reveals differences in crown structure of *Fagus sylvatica* in mixed *vs*. pure European forests," *Forest Ecol. Manage.*, vol. 405, pp. 381–390, 2017, doi: 10.1016/j.foreco.2017.09.043.

[2] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, "YOLOv4: Optimal speed and accuracy of object detection," 2020, *arXiv:2004.10934*.

[3] A. Burt, M. Disney, and K. Calders, "Extracting individual trees from lidar point clouds using treeseg," *Methods Ecol. Evol.*, vol. 10, pp. 438–445, 2019, doi: 10.1111/2041-210X.13121.

[4] L. Cao et al., "Estimating canopy structure and biomass in bamboo forests using airborne LiDAR data," *ISPRS J. Photogramm. Remote Sens.,* vol. 148, pp. 114–129, Feb. 2019, doi: 10.1016/j.isprsjprs.2018.12.006.

[5] A. Chang, Y. Eo, Y. Kim, and Y. Kim, "Identification of individual tree crowns from LiDAR data using a circle fitting algorithm with local maxima and minima filtering," *Remote Sens. Lett.*, vol. 4, no. 1, pp. 29–37, 2013, doi: 10.1080/2150704X.2012.684362.

[6] 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.*, Jul. 2017, pp. 652–660, doi: 10.1109/CVPR.2017.16.

[7] X. Chen, K. Jiang, Y. Zhu, X. Wang, and T. Yun, "Individual tree crown segmentation directly from UAV-borne LiDAR data using the pointnet of deep learning," *Forests,* vol. 12, no. 2, Jan. 2021, Art. no. 131, doi: 10.3390/f12020131.

[8] Y. Chen et al., "Estimation of forest leaf area index using terrestrial laser scanning data and path length distribution model in open-canopy forests," *Agricultural Forest Meteorol.*, vol. 263, pp. 323–333, Dec. 2018, doi: 10.1016/j.agrformet.2018.09.006.

[9] Z. Dong et al., "Registration of large-scale terrestrial laser scanner point clouds: A review and benchmark," *ISPRS J. Photogramm. Remote Sens.*, vol. 163, pp. 327–342, May 2020, doi: 10.1016/j.isprsjprs.2020.03.013.

[10] K. Duan, S. Bai, L. Xie, H. Qi, Q. Huang, and Q. Tian, "CenterNet: Keypoint triplets for object detection," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, Oct. 2019, pp. 6568–6577, doi: 10.1109/ICCV.2019.00667.

[11] C. Hopkinson, L. Chasmer, C. Young-Pow, and P. Treitz, "Assessing forest metrics with a ground-based scanning lidar," *Can. J. Forest Res.*, vol. 34, no. 3, pp. 573–583, Mar. 2004, doi: 10.1139/x03-225.

[12] Q. Hu et al., "RandLA-Net: Efficient semantic segmentation of large-scale point clouds," in *Proc. Comput. Vis. Pattern Recognit. Conf.* [Online]. Available: https://arxiv.org/abs/1911.11236

[13] Z. Hui, Y. Hu, Y. Y. Yevenyo, and X. Yu, "An improved morphological algorithm for filtering airborne LiDAR point cloud based on multi-level kriging interpolation," *Remote Sens.*, vol. 8, no. 1, Jan. 2016, Art. no. 35, doi: 10.3390/rs8010035.

[14] H. Guan et al., "Street-scene tree segmentation from mobile laser scanning data," in *Proc. XXIII ISPRS Congr.*, 2016, pp. 221–225.

[15] Q. Li, P. Yuan, X. Liu, and H. Zhou, "Street tree segmentation from mobile laser scanning data," *Int. J. Remote Sens.*, vol. 41, no. 18, pp. 7145–7162, Sep. 2020, doi: 10.1080/01431161.2020.1754495.

[16] X. Liang et al., "International benchmarking of terrestrial laser scanning approaches for forest inventories," *ISPRS J. Photogramm. Remote Sens.*, vol. 144, pp. 137–139, Oct. 2018, doi: 10.1016/j.isprsjprs.2018.06.021.

[17] X. Long et al., "PP-YOLO: An effective and efficient implementation of object detector," 2020, *arXiv:2007.12099*.

[18] D. Mongus, N. Lukač, and B. Žalik, "Ground and building extraction from LiDAR data based on differential morphological profiles and locally fitted surfaces," *ISPRS J. Photogramm. Remote Sens.,* vol. 93, pp. 145–156, Jul. 2014, doi: 10.1016/j.isprsjprs.2013.12.002.

[19] K. Olofsson, J. Holmgren, and H. Olsson, "Tree stem and height measurements using terrestrial laser scanning and the RANSAC algorithm," *Remote Sens.*, vol. 6, no. 5, pp. 4323–4344, May 2014, doi: 10.3390/rs6054323.

[20] P. Raumonen, E. Casella, K. Calders, S. Murphy, M. Åkerblom, and M. Kaasalainen, "Massive-scale tree modelling from TLS data," in *Proc. PIA15+HRIGI15 – Joint ISPRS Conf.*, Mar. 2015, pp. 189–196, doi: 10.5194/isprsannals-II-3-W4-189-2015.

[21] J. Redmon and A. Farhadi, "YOLOv3: An incremental improvement," 2018, *arXiv:1804.02767*.

[22] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 6, pp. 1137–1149, Jun. 2016, doi: 10.1109/TPAMI.2016.2577031.

[23] M. Rutzinger, A. K. Pratihast, S. O. Elberink, and G. Vosselman, "Detection and modelling of 3D trees from mobile laser scanning data," *Int. Soc. Photogramm. Remote Sens.*, vol. XXXVIII, pp. 520–526, 2010.

[24] K. Tansey, N. Selmes, A. Anstee, N. J. Tate, and A. Denniss, "Estimating tree and stand variables in a Corsican Pine woodland from terrestrial laser scanner data," *Int. J. Remote Sens.*, vol. 30, no. 19, pp. 5195–5209, doi: 10.1080/01431160902882587.

[25] G. Vosselman, "Point cloud segmentation for urban scene classification," *Int. Soc. Photogramm. Remote Sens. Spatial Inf. Sci.*, vol. XL-7/W2, pp. 257–262, 2013, doi: 10.5194/isprsarchives-XL-7-W2-257-2013.

[26] D. Wang, "Unsupervised semantic and instance segmentation of forest point clouds," *ISPRS J. Remote Sens.*, vol. 165, pp. 86–97, 2020, doi: 10.1016/j.isprsjprs.2020.04.020.

[27] D. Wang, M. Hollaus, E. Puttonen, and N. Pfeifer, "Automatic and self-adaptive stem reconstruction in landslide-affected forests," *Remote Sens.*, vol. 8, no. 12, 2016, Art. no. 974, doi: 10.3390/rs8120974.

[28] J. Wang et al., "Individual rubber tree segmentation based on ground-based LiDAR data and faster R-CNN of deep learning," *Forests*, vol. 10, no. 9, 2019, Art. no. 793, doi: 10.3390/f10090793.

[29] L. Windrim and M. Bryson, "Detection, segmentation, and model fitting of individual tree stems from airborne laser scanning of forests using deep learning," *Remote Sens.*, vol. 12, no. 9, 2020, Art. no. 1469, doi: 10.3390/rs12091469.

[30] B. Wu, B. Yu, Q. Wu, Y. Huang, Z. Chen, and J. Wu, "Individual tree crown delineation using localized contour tree method and airborne LiDAR data in coniferous forests," *Int. J. Appl. Earth Observ. Geoinf.*, vol. 52, pp. 82–94, Oct. 2016, doi: 10.1016/j.jag.2016.06.003.

[31] Z. Xi and C. Hopkinson, "Detecting individual-tree crown regions from terrestrial laser scans with an anchor-free deep learning model," *Can. J. Remote Sens.*, vol. 47, pp. 228–242, 2021, doi: 10.1080/07038992.2020.1861541.

[32] Z. Xi, C. Hopkinson, S. B. Rood, and D. R. Peddle, "See the forest and the trees: Effective machine and deep learning algorithms for wood filtering and tree species classification from terrestrial laser scanning," *ISPRS J. Photogramm. Remote Sens.*, vol. 168, pp. 1–16, Oct. 2020.

[33] M. Yadav and B. Lohani, "Identification of trees and their trunks from mobile laser scanning data of roadway scenes," *Int. J. Remote Sens.*, vol. 41, no. 4, pp. 1233–1258, 2020, doi: 10.1080/01431161.2019.1662966.

[34] B. Yang, W. Dai, Z. Dong, and Y. Liu, "Automatic forest mapping at individual tree levels from terrestrial laser scanning point clouds with a hierarchical minimum cut method," *Remote Sens.*, vol. 8, no. 5, 2016, Art. no. 372, doi: 10.3390/rs8050372.

[35] T. Yun et al., "Individual tree crown segmentation from airborne LiDAR data using a novel Gaussian filter and energy function minimization-based approach," *Remote Sens. Environ.*, vol. 256, Apr. 2021, Art. no. 112307, doi: 10.1016/j.rse.2021.112307.

[36] W. Zhang et al., "A novel approach for the detection of standing tree stems from plot-level terrestrial laser scanning data," *Remote Sens.*, vol. 11, no. 2, Jan. 2019, Art. no. 211, doi: 10.3390/rs11020211.

[37] L. Zhong, L. Cheng, H. Xu, Y. Wu, Y. Chen, and M. Li, "Segmentation of individual trees from TLS and MLS data," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 10, no. 2, pp. 774–787, Feb. 2017, doi: 10.1109/JSTARS.2016.2565519.

[38] X. Zhu et al., "Foliar and woody materials discriminated using terrestrial lidar in a mixed natural forest," *Int. J. Appl. Earth Observ. Geoinf.*, vol. 64, pp. 43–50, Feb. 2018, doi: 10.1016/j.jag.2017.09.004.

[39] Z. Yun and G. Zheng, "Stratifying forest overstory and understory for 3-D segmentation using terrestrial laser scanning data," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 14, pp. 12114–12131, 2021, doi: 10.1109/JSTARS.2021.3129312.

[40] K. Itakura, S. Miyatani, and F. Hosoi, "Estimating tree structural parameters via automatic tree segmentation from LiDAR point cloud data," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 15, pp. 555–564, 2022, doi: 10.1109/JSTARS.2021.3135491.

[41] M. Latella, F. Sola, and C. Camporeale, "A density-based algorithm for the detection of individual trees from LiDAR data," *Remote Sens.*, vol. 13, no. 5, 2021, Art. no. 322. [Online]. Available: https://doi.org/10.3390/rs13020322

Lihong Chang received the bachelor's degree in surveying and mapping engineering from the Taiyuan University of Technology, Taiyuan, China, in 2016, and the master's degree in surveying and mapping engineering in 2019 from Wuhan University, Wuhan, China, where she is currently working toward the Ph.D. degree in cartography and geographic information engineering.

Her primary research interest includes computer vision and point cloud processing.

Hongchao Fan received the bachelor's degree in geographic information system from Wuhan University, Wuhan, China, in 2000, the master's degree in geodesy and geoinformatics from the University of Stuttgart, Stuttgart, Germany, in 2006, and the Ph.D. degree from the Technical University of Munich, Munich, Germany, in 2010.

He is currently a Professor with the Norwegian University of Science and Technology, Trondheim, Norway. His main research interests include laser scanning, 3-D city modeling, and photogrammetry.

Dr. Fan is an Associate Editor for the *Arabian Journal of Geosciences*, an Associate Editor for the *Journal of Smart Cities*, and is on editorial boards of the *Journal of Geodesy and Geoinformation* (*Kartographische Nachrichten*).

Ningning Zhu received the bachelor's in surveying and mapping engineering and master's degree in surveying and mapping from Henan Polytechnic University, Henan, China, in 2012 and 2015, respectively, and the Ph.D. degree in photogrammetry and remote sensing from Wuhan University, Wuhan, China, in 2019.

He is currently an Assistant Lecturer with Wuhan University. His primary research is point cloud processing.

Zhen Dong (Member, IEEE) received the bachelor's degree in surveying and mapping engineering, the master's degree in geodesy and surveying engineering, and the Ph.D. degree in photogrammetry and remote sensing from Wuhan University, Wuhan, China, in 2011, 2013, and 2018, respectively.

He is currently an Associate Research Fellow with Wuhan University. His primary research interest includes deep learning on point clouds, smart city, and augmented reality.