forests

MDPI

Article

Stem Detection from Terrestrial Laser Scanning Data with Features Selected via Stem-Based Evaluation

Maolin Chen [1,2], Xiangjiang Liu [1,*], Jianping Pan [1], Fengyun Mu [1,2] and Lidu Zhao [1]

1 School of Smart City, Chongqing Jiaotong University, No. 66, Xuefu Avenue, Nan'an District, Chongqing 400074, China; maolinchen@cqjtu.edu.cn (M.C.); panjianping@cqjtu.edu.cn (J.P.); mfysd@cqjtu.edu.cn (F.M.); zhaolidu@cqjtu.edu.cn (L.Z.)
2 Key Laboratory of Urban Resources Monitoring and Simulation, Ministry of Natural Resources, Shenzhen 518000, China
* Correspondence: xiangjiangliu@mails.cqjtu.edu.cn; Tel.: +86-156-8397-3717

Abstract: Terrestrial laser scanning (TLS) is an effective tool for extracting stem distribution, providing essential information for forest inventory and ecological studies while also assisting forest managers in monitoring and controlling forest stand density. A feature-based method is commonly integrated into the pipelines of stem detection, facilitating the transition from stem point to stem instance, but most studies focus on feature effectiveness from the point level, neglecting the relationship between stem point extraction and stem detection. In this paper, a feature-based method is proposed to identify stems from TLS data, with features selected from stem levels. Firstly, we propose a series of voxel-based features considering the stem characteristics under the forest. Then, based on the evaluation of some commonly used and proposed features, a stem-based feature selection method is proposed to select a suitable feature combination for stem detection by constructing and evaluating different combinations. Experiments are carried out on three plots with different terrain slopes and tree characteristics, each having a sample plot size of about 8000 m^2. The results show that the voxel-based features can supplement the basic features, which improve the average accuracy of stem point extraction and stem detection by 9.5% and 1.2%, respectively. The feature set obtained by the proposed feature selection method achieves a better balance between accuracy and feature number compared with the point-based feature selection method and the features used in previous studies. Moreover, the accuracies of the proposed stem detection methods are also comparable to the three methods evaluated in the international TLS benchmarking project.

Keywords: terrestrial laser scanning (TLS); feature extraction and selection; voxel-based feature; stem detection; forestry

check for updates

Citation: Chen, M.; Liu, X.; Pan, J.; Mu, F.; Zhao, L. Stem Detection from Terrestrial Laser Scanning Data with Features Selected via Stem-Based Evaluation. *Forests* **2023**, *14*, 2035. https://doi.org/10.3390/f14102035

Academic Editors: Huaiqing Zhang, Juan Suárez-Minguez, Qi Chen, Yunsheng Wang, Safa Tharib, Hua Sun, Weipeng Jing, Huaguo Huang, Ting Yun and Meili Wang

Received: 4 August 2023
Revised: 8 October 2023
Accepted: 10 October 2023
Published: 11 October 2023

1. Introduction

Terrestrial laser scanning (TLS) is an effective tool for observing vegetation objects (e.g., tree stems) at the forest holding level because it can acquire detailed three-dimensional (3D) point clouds in a nondestructive way [1,2]. For forest resources inventory, stem detection is the foundation of many applications such as stem mapping [3,4], the estimation of the diameter at breast height (DBH) [5], stem volume prediction [6], and stem reconstruction [7].

There have been many studies retrieving stem information from the TLS point cloud, which can be categorized into model-fitting and feature-based methods. In the model-fitting methods, the raw point clouds are divided into slices along the vertical direction, and model fitting (e.g., circle fitting [8–10], circle-ellipse fitting [11], and cylinder fitting [3,12]) is applied to the sliced points. The point clouds that are successfully fitted to the model will be considered as stem points. However, those methods are easily disturbed by a high stem density and rich understory vegetation, leading to model fitting failure. Therefore, the height and thickness of slices play crucial roles in model-fitting methods [13]. Moreover, before model fitting, image-based methods [14–16] and top-based methods [17,18] were used to extract tree locations.

Feature-based methods are commonly used to directly identify all potential wood or stem points from TLS data. Béland et al. [19] classified wood and leaf points based on intensity information because the intensity of wood points is usually higher than that of leaves. Ma et al. [20] separated photosynthetic and non-photosynthetic components using geometric information and two additional enhanced ground filters, which improved the overall classification accuracy of forest canopies. Zhang et al. [21] proposed a method to extract stem points using the curvature feature of the points and connected component segmentation. A part of the branches and foliage points were removed by using the difference of the surface curvature, and subsequently, the stem points were identified effectively via connected component segmentation. Moreover, the deviation between the z-component of normal vectors is commonly used in the first step of stem and leaf separation because it represents the spatial distribution relative to the vertical direction (i.e., verticality) [22,23]. Wang et al. [24] used the median z-normal value and the planar density to detect the stem locations. Liang et al. [12] used the local geometrical features (i.e., flatness and normal direction) to identify the stem points. Xia et al. [25] adopted two-scale classification based on the dimensionality features to extract the candidate stem points. Additionally, the point cloud is translated into a voxel structure and the flatness saliency feature is calculated to recognize the stem points [26].

Moreover, machine learning algorithms are confirmed for leaf–wood classification by using abundant geometric information. Zhu et al. [27] extracted a series of local radiometric and geometric features derived from TLS point clouds and separated wood and leaf points using the Random Forest (RF) model. The overall classification accuracy is between 80% and 90%. Chen et al. [28] also applied the Support Vector Machine (SVM) classifier and combined multiple features, such as dimensionality features, normal vectors, and intensity values. Hackel et al. [29] and Becker et al. [30] constructed 16- and 15-geometric feature Random Forest training models, with classification accuracies of 90% and 84%, respectively. Moorthy et al. [31] used radially bounded nearest neighbors at multiple spatial scales to classify wood and foliage in an RF model. There were 30 features applied, with an overall average accuracy of 94.2%. However, research shows that more features do not mean a higher accuracy in the classification [32].

In summary, most previous feature-based methods have achieved satisfactory point-based results. Then, the individual stem can be detected using segmentation and clustering based on prior knowledge (e.g., tree growth direction) [4,33]. However, there is still room for improvement and discussion. Firstly, although voxel-based features have been proven to be effective for object recognition in urban scenes [34–37] and have been used to extract stem points with a series of predefined segmentation rules [38,39], their effectiveness has been discussed less under a classification framework in terms of stem extraction results. Secondly, the combination of features, rather than simply integrating features, plays a key role in classification [38,40]. In a forest environment, particularly, factors such as variations in the point cloud density and occlusion can render the features ineffective. Therefore, constructing reasonable feature combinations is a critical aspect of improving the stem detection result. Thirdly, the performance of different feature combinations is commonly evaluated based on point-based results in urban scenes [32,41,42]. However, the utilization of a forest point cloud is typically centered around stem instances, and there has been limited discussion in previous studies regarding the assessment of various feature combinations in the context of stem-based results.

Considering the points mentioned above, the target of this study is to detect stems using a feature-based method and analyze the potential relationship between different feature combinations and stem-based results. Our two main contributions are (1) proposing a series of voxel-based features to identify stem points and detect stems from TLS data considering the stem characteristics under a forest and (2) developing a feature selection method by analyzing the effectiveness of different feature combinations in terms of stem-based metrics.

This article is structured as follows: In Section 2, we provide a comprehensive introduction of the data. Section 3 describes the proposed approach. Following that, Section 4 presents the experimental results. In Section 5, we engage in a discussion of the approach. Lastly, in Section 6, we conclude this article.

2. Data Acquisition and Introduction

Two groups of datasets are used in this study. The first dataset was obtained from three plots with different topographic and tree characteristics (Figure 1). Plot 1 was collected in a camphor forest at Wuhan university (30.54° N, 114.36° E), China, characterized by a flat terrain, straight tree stems, and sparse understory vegetation. Plot 2 was obtained from a sloping mixed forest in Luojia hill (30.59° N, 114.29° E), China, primarily consisting of Masson's pine, oak and cypress trees. It has a higher density with smaller trees, denser understory vegetation, and more irregular and diverse stem forms. Plot 3 was collected in a mountainous region in Shennongjia (30.07° N, 112.32° E) in China, where the terrain fluctuates greatly and is mainly covered by *Euptelea pleiospermum*. In this plot, the trees start branching from a height close to the ground and have diverse growth directions. All data were acquired in single-scan mode using Riegl VZ-400 (Riegl GmbH, Horn, Austria) with an angular density of 0.04°, and the largest distance to scanner position was more than 100 m. Detailed information of the three plots is shown in Table 1. The second dataset was obtained from the international TLS benchmarking project organized by the Finish Geospatial Research Institute, including six forest plots obtained using both single- and multi-scan format. These plots are situated in the southern boreal forests of Evo, Finland, covering various tree species, stem densities, developmental stages, and understory vegetation richness. Based on these characteristics, the six plots were categorized into three complexity levels (i.e., easy, medium, and difficult), each with a fixed size of 32×32 m^2. Further details regarding the comprehensive description of the benchmark datasets can be found in [43]. All experiments were performed with MATLAB R2018b on a computer with Intel Core i7-10750H CPU and 32 GB RAM.

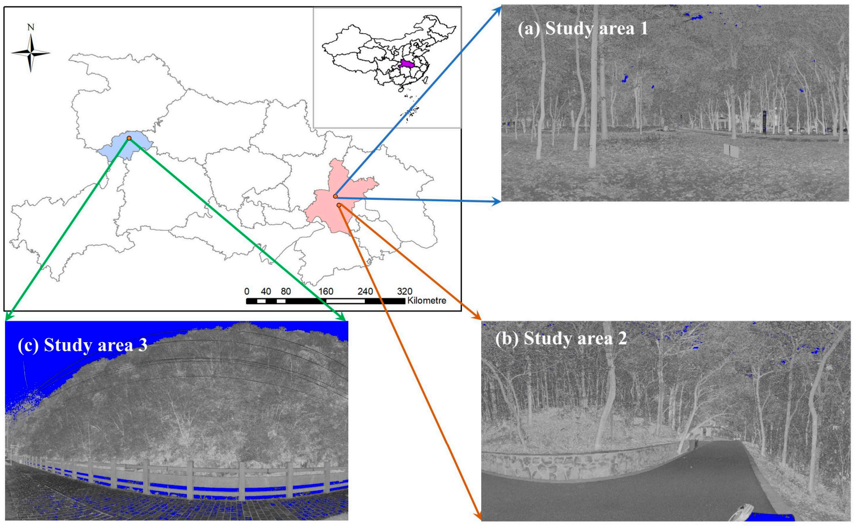
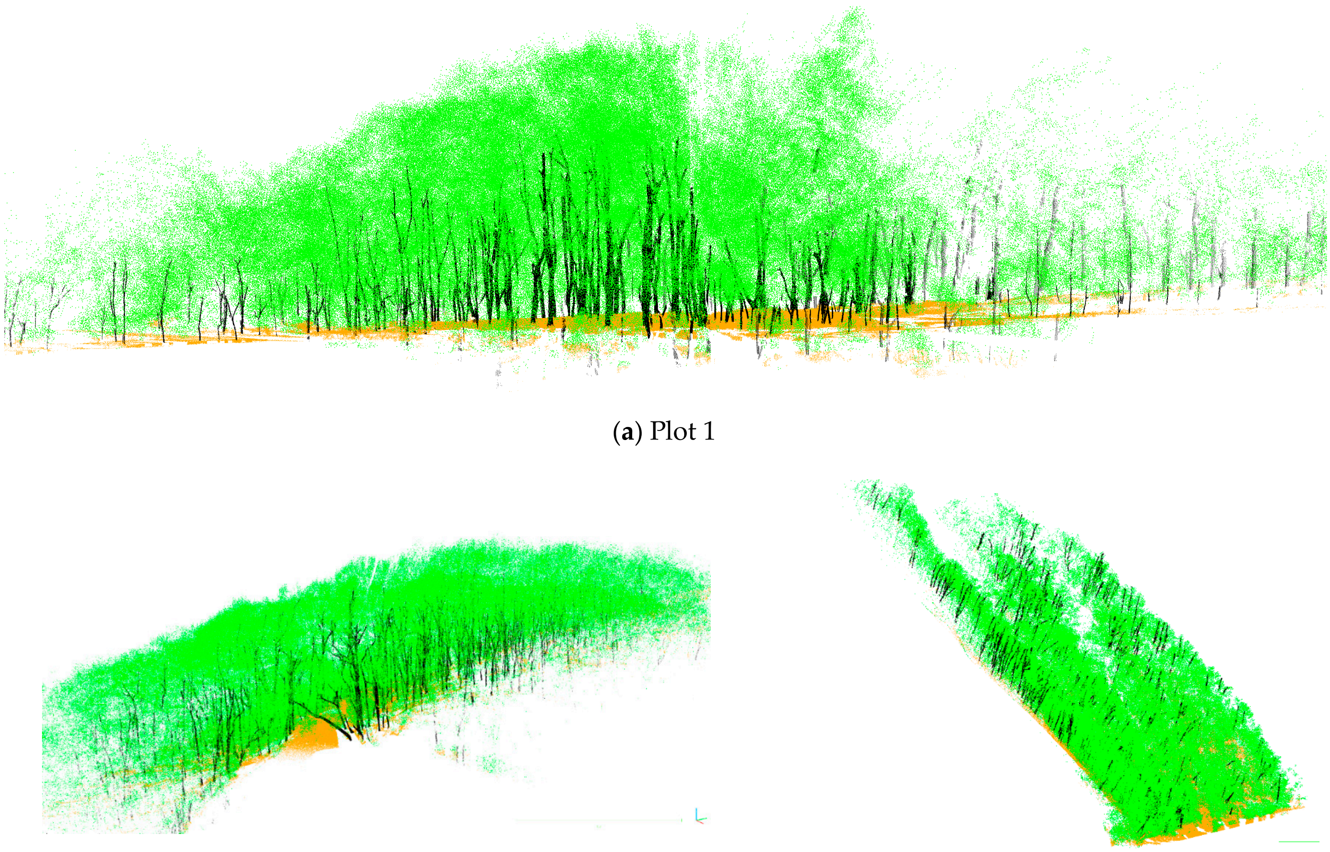


Figure 1. The location of the study area of first dataset.

But as point-level accuracy is also validated in this study, point-level reference data are also required. Below is the process of manually labeling each point in this study. We manually divided all the points into three classes: foliage, stem, and ground. Initially, the ground points were extracted via CSF filtering, and the misclassified points were then refined through manual inspection. Subsequently, we separated the stem points from the remaining points, with only the trunk, primary, and secondary branches labeled as stem points in this study, as shown in Figure 2.

Table 1. Detailed information of the three plots in the first dataset.

Plot ID	Plot Size (m^2)	Sum of Trees	DBH (cm)	Tree Height (m)	Slope (°)	Point Spacing (m)	Point Number			
							All	Stem	Ground	Foliage
1	120 × 70	287	27.5 ± 6.9	14.9 ± 2.5	~2	0.001~1.175	23,794,665	1,996,887	8,056,326	13,741,452
2	140 × 60	386	21.8 ± 5.0	11.1 ± 4.6	~17	0.001~0.935	23,332,788	2,179,648	6,692,757	14,460,383
3	140 × 60	480	12.8 ± 4.2	7.6 ± 2.3	~42	0.001~1.425	7,568,384	508,853	2,112,451	4,947,080



(**a**) Plot 1

(**b**) Plot 2

(**c**) Plot 3

Figure 2. Illustration of manually labeled point cloud in first dataset. Green: foliage points. Black: stem points. Orange: ground points.

3. Methods

The proposed method consists of four major steps (Figure 3): feature extraction, point classification, stem detection, and feature selection. The goal of the first two stages is to determine the subset of probable candidate stem points from the entire input. This is achieved via supervised classification with a series of features, including some commonly used features and the proposed features. Sections 3.1 and 3.2 describe these procedures in detail. As outlined in Section 3.3, a clustering-based point cloud segmentation framework is applied to individual stem extraction. In the final section of the method, the features are evaluated and selected via the stem-based results after the feature importance ranking based on the point classification results.

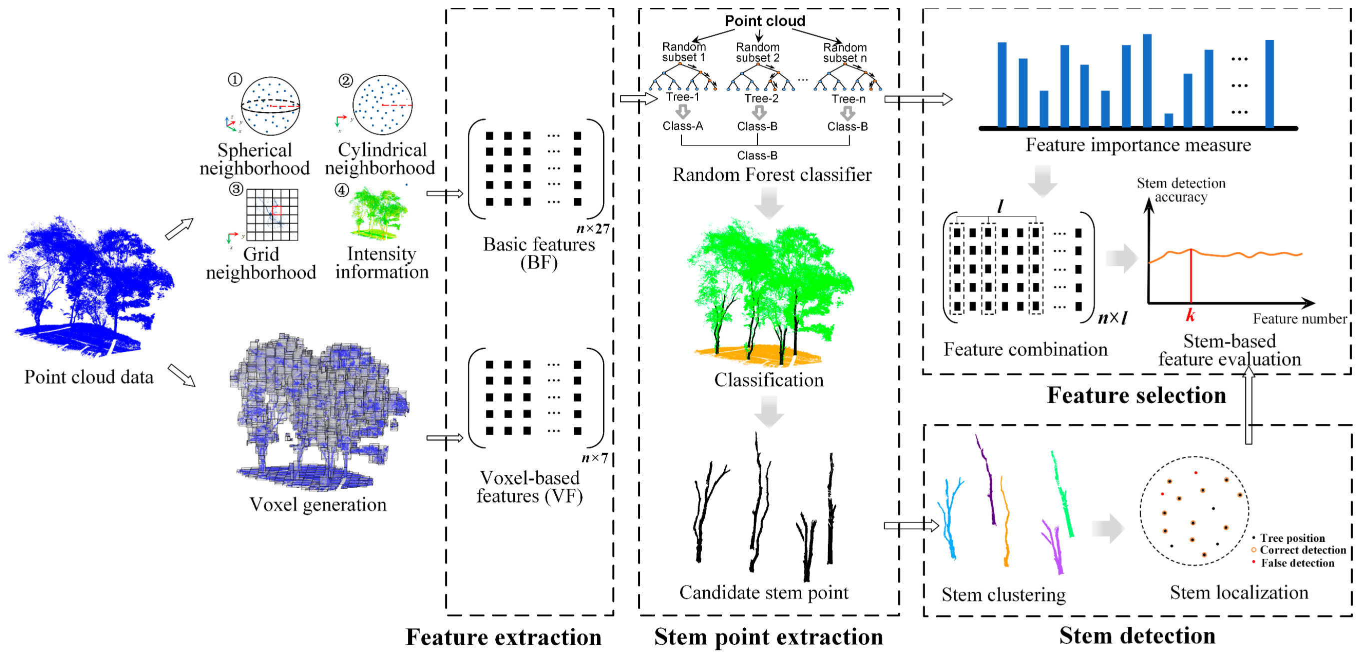


Figure 3. The general framework of this study.

3.1. Feature Extraction

We extract candidate stem points using the classification method in this study. Constructing reasonable features is crucial for obtaining satisfying classification results [44,45], and we select the features proposed in previous studies [27,32,46,47] as basic features in this study. As the aim is to provide a group of basic features for the following proposed feature selection strategy, this study does not attempt to cover all ever-used point features. Additionally, feature correlation can be recognized partially by the Recursive Feature Elimination of Random Forest [48] in our strategy. Then, we follow the feature construction and classification framework in [41,49], classifying the basic features (BFs) into 3D features, 2D features, grid features, and intensity-based features, and voxel-based features (VFs) are proposed.

3.1.1. Basic Features (BFs)

1. Three-dimensional features

Each 3D point and its k nearest points form a local neighborhood, and we optimize k for each point to obtain an optimal neighborhood according to the principle of the minimum eigenentropy [41]. In the search process for the optimal neighborhood, the three eigenvalues, λ_1, λ_2, and λ_3 ($\lambda_1 > \lambda_2 > \lambda_3$), are obtained by performing a principal component analysis (PCA) [50], and the eigenentropy E_λ is calculated using Equation (1). Subsequently, by comparing the eigenentropy across different neighborhoods, the neighborhood associated with the minimum eigenentropy is chosen as the optimal neighborhood for each point. Figure 4 illustrates the process of neighborhood selection. Based on the eigenvalues of the optimal neighborhood, a variety of geometric 3D features are defined to represent the spatial distribution of the 3D point cloud within the neighborhood. Table 2 summarizes the first set of 3D features.

$$E_\lambda = -\lambda_1 \ln(\lambda_1) - \lambda_2 \ln(\lambda_2) - \lambda_3 \ln(\lambda_3) \quad (1)$$

2. Two-dimensional features

The extraction process of 2D features is similar to that of 3D features. First, the 2D point and its k closest neighbors are obtained based on the cylinder-based neighborhood. Then, the eigenvalues, $\lambda_{1,2D}$ and $\lambda_{2,2D}$ ($\lambda_{1,2D} > \lambda_{2,2D}$), are calculated by constructing the second-order covariance matrix. From its eigenvalues, the sum of eigenvalues and some basic properties of local 2D neighborhood can be calculated and exploited as 2D features, as shown in Table 3.

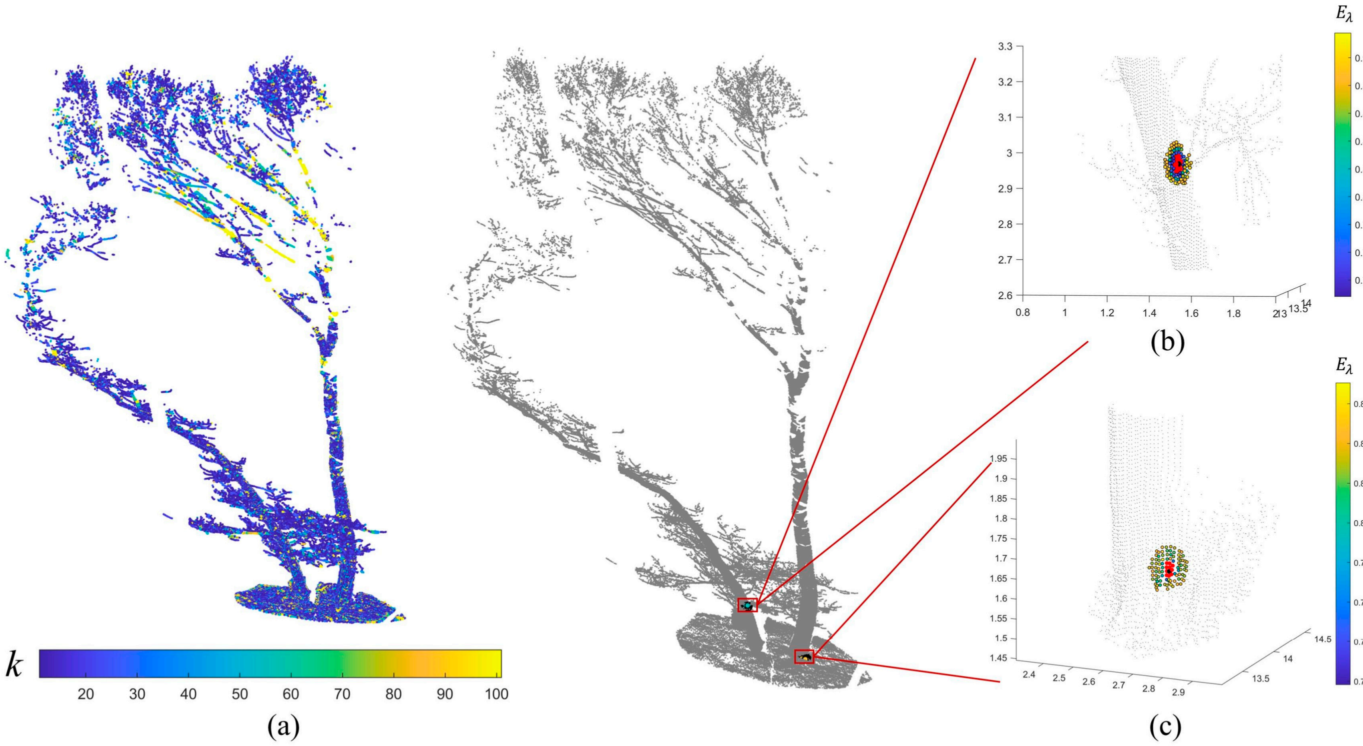


Figure 4. Illustration of the neighborhood selection. (**a**) Neighborhood selection result (colored with the selected k value) on sample data. (**b**) The neighbors p_k ($k \in [10, 100]$, $\Delta k = 1$) of one stem point (black dot) close to foliage. (**c**) The neighbors p_k ($k \in [10, 100]$, $\Delta k = 1$) of one stem point (black dot) close to ground. The color of each neighbor p_k corresponds to the value of eigenentropy E_λ computed on the smallest neighborhood containing p_k. The eigenentropy first decreases until the neighborhood reaches the edge of the stem (red points show the range of optimal neighborhood), and then increases and reaches its maximum when a different object (e.g., foliage and ground) is aggregated within the neighborhood.

3. Grid features

The grid feature is formed by projecting the scanning area into a regular grid in the XY plane. For the points in the XY plane, the regular grid is used as a special neighborhood for feature extraction, which can simplify the calculation related to the spatial relationship of the point cloud. The grid size is set to 1 m by considering the stem spacing in the sample plot, and then the following features are calculated according to the points in each grid:

- N_{grid}: the point number in each grid.
- Δ_H: the maximum height difference of point cloud in each grid.
- σ_H: the standard deviation of point height of each grid.
- Nor_z: the z-value difference between each point and the lowest point in each grid.

4. Intensity-based feature

The intensity information can be used to differentiate wood and leaf points because the intensity of leaf points is usually darker than that of wood. In this paper, the intensity-based feature is labeled as I.

Table 2. Three-dimensional features extracted from the point cloud.

No.	Symbol	Feature	Definition
1	L_λ	linear saliency	$(\lambda_1 - \lambda_2)/\lambda_1$
2	P_λ	planar saliency	$(\lambda_2 - \lambda_3)/\lambda_1$
3	S_λ	volumetric saliency	λ_3/λ_1
4	E_{3D}	Shannon entropy	$-L_\lambda \ln L_\lambda - P_\lambda \ln P_\lambda - S_\lambda \ln S_\lambda$
5	E_λ	eigenentropy	$-\sum_{n=1}^{3} \lambda_n * \ln \lambda_n$
6	n_x	x-component of the normal vector NV [1]	NV (1)
7	n_y	y-component of the normal vector NV	NV (2)
8	n_z	z-component of the normal vector NV	NV (3)
9	O_λ	structure tensor omnivariance	$(\lambda_1 * \lambda_2 * \lambda_3)^{1/3}$
10	A_λ	structure tensor anisotropy	$(\lambda_1 - \lambda_3)/\lambda_1$
11	Sum_λ	sum of eigenvalues	$\sum_{n=1}^{3} \lambda_n$
12	C_λ	change in curvature	$\lambda_3/(\lambda_1 + \lambda_2 + \lambda_3)$
13	V_p	verticality	$1 - n_z$
14	r_{knn}	radius of the spherical neighborhood	\
15	D_{3D}	local point density	$3(k+1)/\left(4\pi r_{knn}^3\right)$

[1] $NV \in \mathbb{R}^3$ denotes the normal vector, and k is the suitable number of closest neighbors for each given 3D point.

Table 3. Seven 2D features extracted from the point cloud.

No.	Symbol	Feature	Definition
1	$L_{\lambda,2D}$	linear saliency of local 2D neighborhood	$(\lambda_1 - \lambda_2)/\lambda_1$
2	$P_{\lambda,2D}$	planar saliency of local 2D neighborhood	λ_2/λ_1
3	E_{2D}	Shannon entropy of 2D neighborhood	$-L_\lambda \ln L_\lambda - P_\lambda \ln P_\lambda$
4	$E_{\lambda,2D}$	eigenentropy of 2D neighborhood	$-\sum_{n=1}^{2} \lambda_{n,2D} * \ln \lambda_{n,2D}$
5	$O_{\lambda,2D}$	structure tensor omnivariance of 2D neighborhood	$(\lambda_1 * \lambda_2)^{1/2}$
6	D_{2D}	local 2D point density	$(k+1)/\left(\pi r_{knn,2D}^2\right)$ [1]
7	$Sum_{\lambda,2D}$	sum of eigenvalues of 2D neighborhood	$\sum_{n=1}^{2} \lambda_{n,2D}$

[1] $r_{knn,2D}$ is radius of the circular neighborhood defined by a 2D point and its k closest neighbors.

3.1.2. Voxel-Based Features (VF)

Voxelization is an effective way to organize and structure 3D point clouds. In our study, we treat a voxel as a form of a neighborhood instead of directly using it to detect stem points. Firstly, we establish cubical voxels over the whole point cloud, as shown in Figure 5b, and only retain voxels containing at least one point to reduce memory consumption and improve traversal speed, as shown in Figure 5c. Then, the voxel is converted to the axis-aligned bounding box (AABB), as shown in Figure 5d, and the voxel-based features (VFs) are extracted based on both points in each voxel and the voxel size change.

Each retained voxel is recoded as V_k, containing m points $\{p_n = (x_n, y_n, z_n) | p_n \in V_k, n = 1, 2, \ldots, m\}$. Similar to the calculation of basic features, we use the covariance matrix $\sum$ to calculate the normal vector of points within each voxel, which is defined as

$$\sum = \frac{1}{m} \sum_{n=1}^{m} (p_n - \overline{p})(p_n - \overline{p})^T \tag{2}$$

where $\overline{p} = (\overline{x}, \overline{y}, \overline{z})$ is the coordinate of the center of the point within voxels.

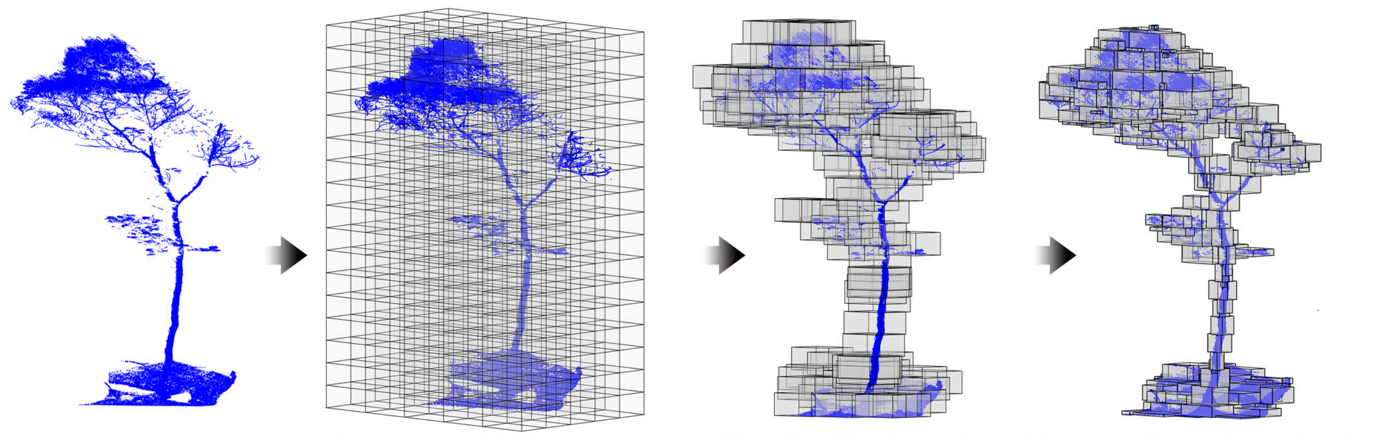


Figure 5. The axis-aligned bounding box (AABB) is formed according to the 3D point in each voxel.

After performing the eigenvalue decomposition of matrix $\sum$ in Equation (2), the process yields three distinct eigenvectors along with their corresponding eigenvalues. Among these eigenvectors, the one associated with the smallest eigenvalue is referred to as the normal vector. The different distributions of leaf, stem, and ground points can be captured by the three components, v_x, v_y, v_z, of normal vector based on voxel space, as shown in Figure 6b–d. This difference in spatial distribution makes it possible to separate stem points from other points. As shown in Figure 6c, the stem points have a dominant direction of normal vector, which is indicated by the v_x or v_y components of the XY plane rather than v_z. Similarly, the ground point has a prominent value of v_z (Figure 6d). Leaf points, on the other hand, have similar values for both v_y and v_z (Figure 6b).

In addition, VFs are extracted based on the size difference between the initial voxel and the formed AABB. The bottom area S_{AABB} and the height h_{AABB} are calculated according to the points of each voxel V_k as follows:

$$S_{AABB} = (max(x_n) - min(x_n)) \times (max(y_n) - min(y_n)) \tag{3}$$

$$h_{AABB} = max(z_n) - min(z_n) \tag{4}$$

where x_n, y_n, and z_n are the x-, y-, and z-axis coordinate values of point p_n, $p_n \in V_k$, $n = 1, 2, \ldots, m$. Then, the bottom area ratio R_S and the height ratio R_h can be calculated.

$$R_S = S_{AABB}/S_{voxel} \tag{5}$$

$$R_h = h_{AABB}/h_{voxel} \tag{6}$$

where S_{voxel} and h_{voxel} indicate the bottom area and height of the initial voxel, which can be calculated using our predefined voxel size.

Figure 6e–g show the differences between the foliage, stem, and ground points. Since the foliage points are usually scattered and evenly distributed in different directions, the size change is relatively small, and both R_S and R_h are almost equal to 1 (Figure 6e). For voxels containing stem points, the height ratio R_h is close to 1, while the bottom area ratio R_S is much smaller (Figure 6f), because the stem is shaped like a slender pole. Conversely, the ground points cover almost the whole bottom part of the voxel, while the height ratio R_h is much smaller (Figure 6g). Moreover, we also calculate the mean intensity and variance of intensity values of the 3D points within each voxel. Finally, the VF is assigned to each 3D point in the voxel. Table 4 summarizes the VFs and depicts the difference between each class using a radar chart.

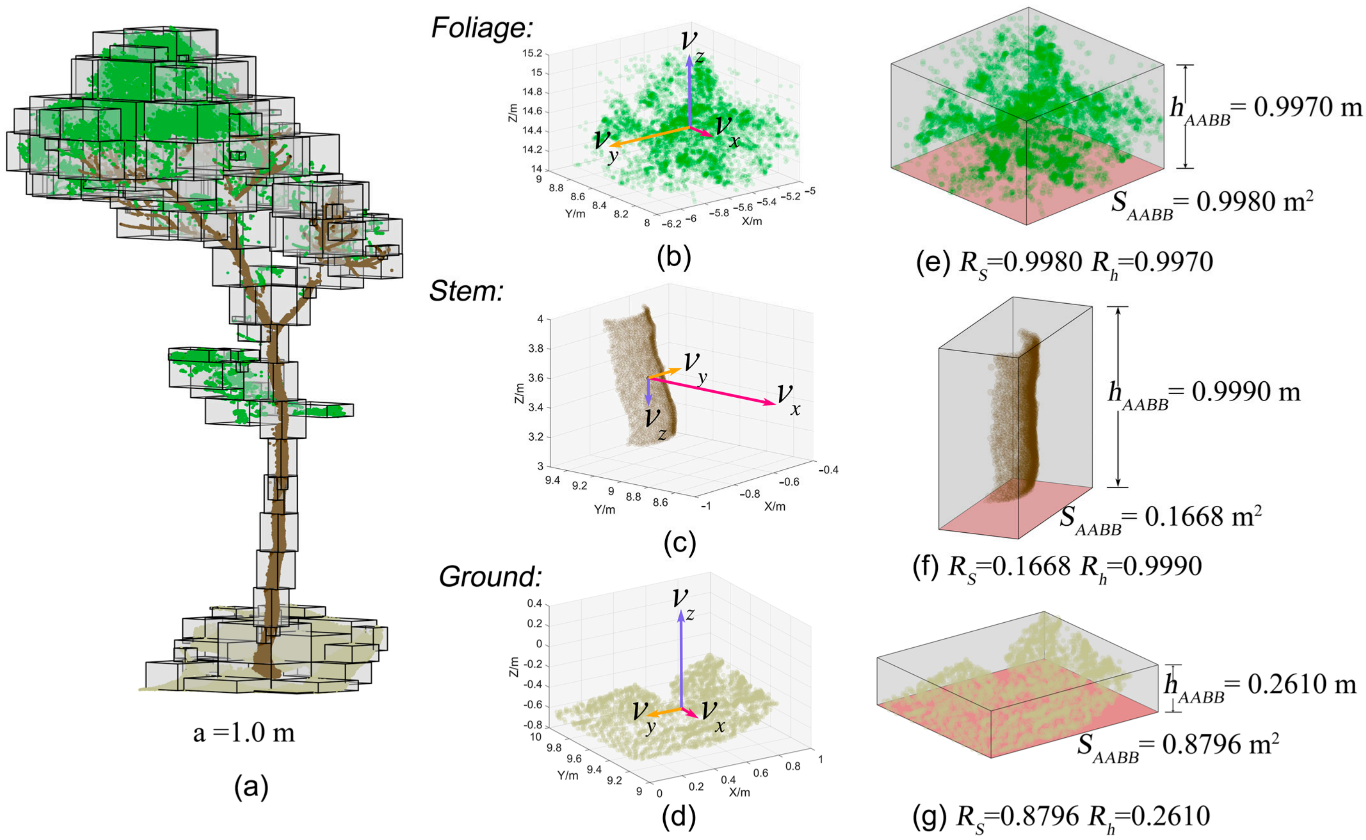
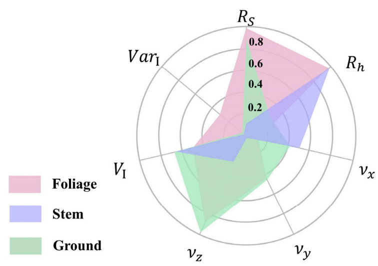


Figure 6. Illustration of (**a**) a tree example with the initial voxel size of 1 m, the three components (v_x, v_y, v_z) of normal vector to represent the different spatial distributions of (**b**) foliage, (**c**) stem, and (**d**) ground points within a voxel, and (**e**–**g**) the height ratio Rh and bottom area ratio RS calculated using the height hAABB and bottom area SAABB of AABBs.

Table 4. Voxel-based features and the difference between each class.

Voxel-Based Feature	Description [1]
R_S	The bottom area ratio
R_h	The height ratio
v_x, v_y, v_z	Three components of normal vector in voxel V_k
V_I	Mean intensity value of constituting 3D points in voxel V_k
Var_I	Variance of intensity values within voxel V_k



[1] The radar chart illustrates variations in the normalized values of features among different classes; distinguishing the differences between these classes is crucial for classification.

3.2. Classification and Feature Importance Evaluation

In this paper, the Random Forest (RF) classifier is used for the stem point identification based on the aforementioned features [50]. The accuracy of stem point extraction is evaluated using Equations (7)–(9).

$$Precision = \frac{TP}{TP + FP} \tag{7}$$

$$Recall = \frac{TP}{TP + FN} \quad (8)$$

$$F1 - score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (9)$$

where TP is the number of correctly classified stem points, FP is the number of wrongly identified stem points, and FN is the number of stem points that are wrongly labeled as other classes. RF is chosen as it gives a good trade-off between accuracy and computational cost. In addition, it has also been proven to be successfully applied to wood–leaf classification from forest TLS point clouds [31,51]. Meanwhile, RF can also be used for feature selection in the workflow of point classification based on the Recursive Feature Elimination (RFE) algorithm [52]. All features are ranked by the decision tree-based strategies, and the Gini index of each feature can be calculated using the average of all decision trees in the RF classifier. Through this process, a Random Forest can estimate the importance of each feature for the classification task.

To train the RF model, 1000 training samples are randomly selected from three classes, respectively. Moreover, all features are used as inputs in the RF classifier. Note that a unity-based normalization is used to bring the values of each dimension into the range [0, 1] before being applied to the Random Forest classifier.

3.3. Stem Detection

3.3.1. Candidate Stem Recognition

Candidate stem points can be obtained after the classification. Due to the similarity of point features and the uncertainty of noise, there are still non-stem points (mainly leaf points). However, the residual non-stem points are sparser than tree stem points after classification. Thus, the Euclidean clustering algorithm is used to generate stem point clusters and exclude non-stem points [25,28]. If the distance between any two points is less than d, they are labeled as the same cluster; otherwise, they are divided into two different clusters. Generally, the point number of stem clusters is much larger than that of the remaining non-stem clusters. Therefore, we remove the clusters with point number less than Nc.

Our situation is similar to that in [28], in which the maximum scanning distance of the scanner is more than 100 m, and the point density decreases with the increase in distance. In this case, a fixed threshold may not be able to generate appropriate stem clusters and filter small clusters. Therefore, according to the method used in [28], the thresholds of d and Nc are generated adaptively, based on the distance. The threshold d can be calculated as

$$d = N * d_p * \rho_{angle} \quad (10)$$

where d_p represents the distance from the current point to the scanner position, and ρ_{angle} represents the angular resolution in radians. Therefore, the threshold d is expressed as N times the arc length formed by two adjacent scanning beams.

The adaptive filtering threshold can be obtained using the following equation:

$$Nc = (D_{min} * \gamma * H_{min}) / \left(\rho_{angle} * d_c\right)^2 \quad (11)$$

where D_{min} and H_{min} are the preset minimum stem diameter and height, respectively. d_c is the average distance from the point in the target cluster to the scanner position, and γ is the occlusion rate, which represents the invisibility or incompleteness of the stem, expressed as a percentage and the value between 0 and 1. According to the characteristics of the trees in different plots, D_{min} and H_{min} are set to 0.2 and 10 m (plot 1), 0.05 and 8 m (plot 2), and 0.05 and 4 m (plot 3), respectively.

3.3.2. Calculation of Stem Position

In order to calculate the stem position, we use a cylinder model based on the Random Sample Consensus algorithm (RANSAC) [53] to fit the lowest slice of clusters. The cylindrical fitting needs to meet the following two conditions. First, the diameter of a cylinder should be within the range of [0.05, 0.5]. This range is intuitively related with the size of the trees in a specific plot and it will work well as long as the distribution range of the diameter is covered for most trees. Thus, it can be determined based on prior knowledge about the tree species, a random single tree inspection from the point cloud, and on-site sampling. Second, the angle between the cylindrical axis and the z-axis should be less than 8°. It is suitable for most cases, as most trees grow vertically. The intersection point of the cylindrical axis vector with the triangle irregular network (TIN) model constructed from ground points is calculated as the stem position, as shown in Figure 7.

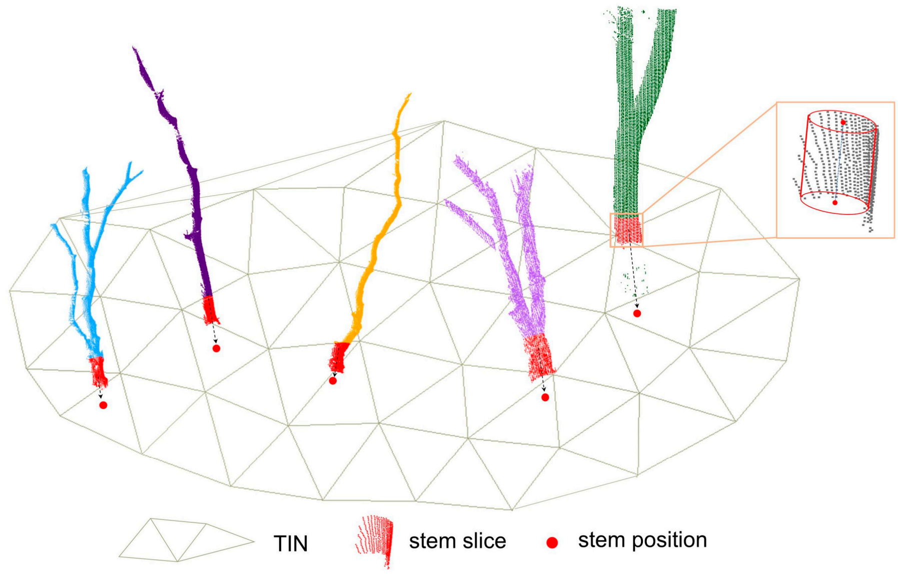


Figure 7. Calculation of stem position. Each stem were assigned random colors.

3.4. Stem-Based Feature Selection

Feature combination is commonly evaluated based on point-based classification results, but the basic application is usually based on stems in forest TLS data. Thus, we propose a feature selection method based on stem detection results in this study. The main steps are as follows:

(1) Data preparation. A sample scan is picked from a plot, and sample points are manually selected for RF training, as described in Section 3.2. Feature importance ranking can be obtained using an RF classifier with the Recursive Feature Elimination (RFE) algorithm. Then, stem positions are marked manually as reference for stem-based result evaluation.
(2) Feature combination construction. We sort all features in a decreasing order of feature importance and iteratively add them to form different feature combinations. Each combination is used for stem point extraction and stem detection.
(3) Feature combination evaluation. Stem-based accuracy is calculated for each combination based on the reference data obtained in (1).

$$Correctness = \frac{n_{match}}{n_{extr}} \tag{12}$$

$$Completeness = \frac{n_{match}}{n_{ref}} \tag{13}$$

$$IoU = \frac{n_{match}}{n_{ref} + n_{extr} - n_{match}} \tag{14}$$

where n_{match} is the number of found reference stems, n_{extr} is the number of extracted stems, and n_{ref} is the number of reference stems. The feature combination with the highest IoU is regarded as the optimal, as shown in Figure 8.

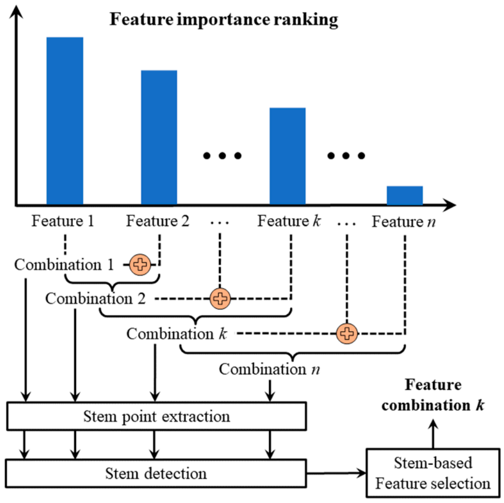


Figure 8. The flowchart of the proposed feature selection method.

4. Results

4.1. Performance of Voxel-Based Features

In order to analyze the performance of voxel-based features, we compare the results with and without voxel-based features, as shown in Tables 5 and 6. Thus, two groups of feature combinations are used in this test. The first only consists of the basic features described in Section 3.1.1, labeled as BF in Tables 5 and 6. The second consists of both basic and voxel-based features, labeled as BF-VF in Tables 5 and 6. The voxel size for BF-VF is set as $x \in \{0.2\text{ m}, 0.4\text{ m}, 0.6\text{ m}, 0.8\text{ m}, 1.0\text{ m}, 1.2\text{ m}, 1.4\text{ m}\}$, according to the distance of the adjacent stems and the average DBH of the stem in the scene. The result of BF-VF is averaged by different voxel sizes.

Table 5. Results of stem point extraction with and without the voxel-based features.

Plot ID	Actual Quantity	Feature Set	Estimated Quantity	Correctly Estimated Quantity	Precision (%)	Recall (%)	F1-Score (%)
1	1,996,887	BF	2,748,648	1,769,377	64.41	88.61	74.58
		BF-VF	2,481,683	1,833,304	**73.91**	**91.81**	**81.88**
2	2,179,648	BF	1,915,221	1,132,381	59.13	51.95	55.31
		BF-VF	2,012,364	1,402,336	**69.79**	**64.34**	**66.86**
3	508,853	BF	605,041	256,065	42.34	50.32	45.98
		BF-VF	577,693	301,287	**52.18**	**59.21**	**55.46**

The point-based results are shown in Table 5. In order to facilitate an objective comparison, the accuracy of each feature combination is averaged over 10 loops since the result of the RF classifier may vary slightly for each run. It shows that the voxel-based features significantly improve the result of the stem point extraction, with the F1-score increasing by 7.3%, 11.6%, and 9.5% for plots 1, 2, and 3, respectively. In addition, with the increase in the complexity of the sample plot, the accuracy of stem point extraction gradually decreases. Figure 9 shows the detailed comparison of the classification results. It can be seen that more continuous stem (the red boxes in Figure 9a,c) and foliage points (the red box in Figure 9b) can be obtained by including voxel-based features. However, within the blue boxes in Figure 9b, we observe less satisfactory classification results, particularly in areas characterized by thin stems and the presence of foliage. The occlusion of the foliage in these areas diminishes the number of identifiable stem points, which, in turn, affects the effectiveness of voxel-based features.

Table 6. Results of stem detection with and without the voxel-based features.

Plot ID	n_{ref}	Feature Set	n_{extr}	n_{match}	Correctness (%)	Completeness (%)	IoU (%)
1	287	BF	322	250	77.64	**87.11**	69.64
		BF-VF	317	248	**78.53**	86.56	**70.00**
2	386	BF	478	352	73.64	91.20	68.75
		BF-VF	462	353	**76.57**	**91.41**	**71.38**
3	480	BF	537	392	73.00	81.67	62.72
		BF-VF	545	398	**73.01**	**82.83**	**63.41**

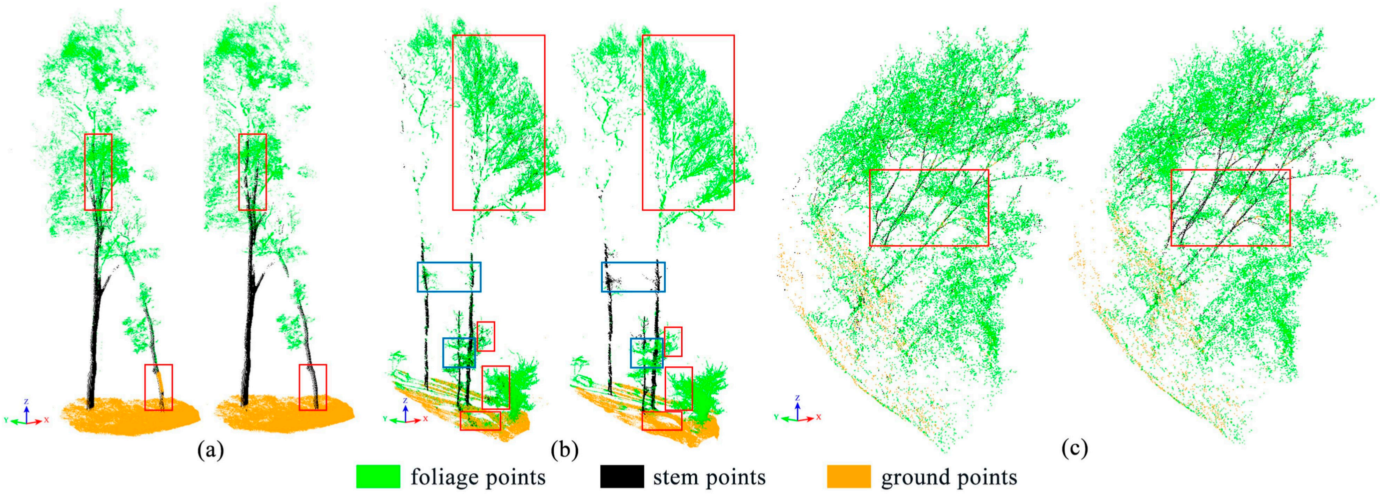


Figure 9. Detailed comparison of classification results without (left) and with (right) voxel-based features at (**a**) plot 1, (**b**) plot 2, and (**c**) plot 3.

Table 6 shows the results of stem detection generated on the basis of extracted stem points. The IoU increases by 0.4%, 2.6%, and 0.7% for plots 1, 2, and 3, respectively, when voxel-based features are involved. This shows that the voxel-based features are also helpful in stem detection. It should be noticed that, although more stem points are extracted by BF-VF in plot 1 (recall of 91.81% versus 88.61% in Table 5), fewer stems are detected with the completeness of 0.8656 versus 0.8711 in Table 6. This is mainly because a certain stem segment can be sufficient for stem detection, even the points on the stem surface are not completely extracted.

In order to evaluate the effect of the voxel size, we make an insight into the results in terms of different voxel sizes, as shown in Figure 10. It shows that including voxel-based features can improve the result of stem point extraction with all tested parameters, while the improvement in stem-based results differs among different voxel sizes (Figure 10a). A voxel size of [0.2 m, 0.8 m], including voxel-based features, can improve the stem-based results stably, but both the best and worst results occur when the voxel size is larger than 1 m (Figure 10b). Additionally, a better stem point extraction result may not achieve a better stem detection result, which is similar to the results in Tables 5 and 6. In total, the proposed voxel-based features are beneficial to improve both point- and stem-based results and the improvement is stable in different parameter settings.

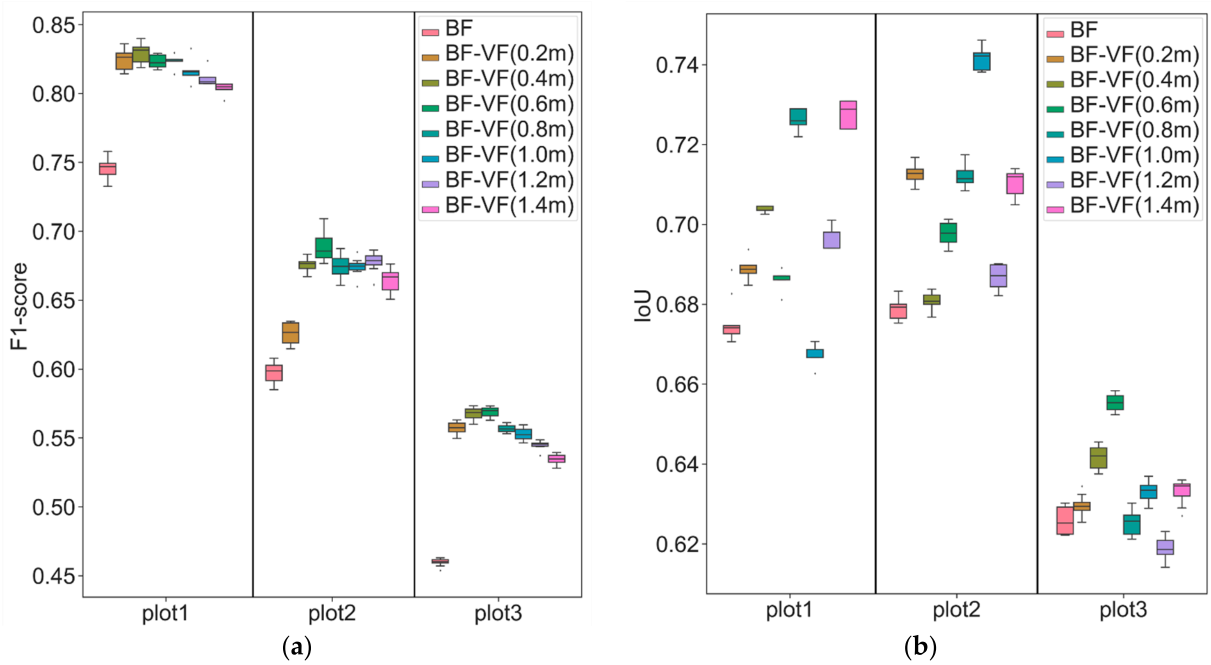


Figure 10. The results of (**a**) stem point extraction and (**b**) stem detection in terms of different voxel sizes. BF: basic features; BF-VF(x): basic features and voxel-based features with voxel size of x.

4.2. Effects of Feature Combination on Stem Detection

As shown in Figure 11, we use the Random Forest method to evaluate the importance of each feature for different plots, and the importance value is averaged over different voxel sizes. It is noted that there are 2–4 voxel-based features among the top 5 features ranked by importance in different plots, which helps to explain why voxel-based features can improve the accuracy of stem point extraction. It is worth mentioning that the feature importance ranking in plot 3 is different from those in plots 1 and 2, which is mainly reflected in the features related to vertical distribution, including n_z, V_p, R_h, and v_z. The probable reason is the large slope in plot 3 and the diverse stem direction.

The point-based feature selection method is commonly used to evaluate features [54,55], so it is used for comparison with the proposed method. To generate different feature combinations, the order of feature importance in Figure 11 is used to successively generate stem-based results with one additional feature added per iteration. The results of stem point extraction and stem detection under different feature combinations are depicted in Figure 12. It shows that with the increase in the feature number, the accuracy of stem point extraction gradually improves at the beginning and finally tends to be stable, which is similar to the conclusion in many previous studies [32,38,42]. And the stem-based accuracy shows the similar trend as a point-based result. However, the result of stem detection is not fully synchronized with the point-based result. Therefore, according to the point-based feature selection method, the optimal feature subsets in the three plots are composed of 26-, 22-, and 19 best-ranked features, respectively, while the first 9, 12, and 15 features will be selected using the proposed method. Compared to the proposed method, the features selected using the point-based feature selection method did not yield higher stem detection results, which further proves that extracting more stem points may not be a necessary condition to stem detection.

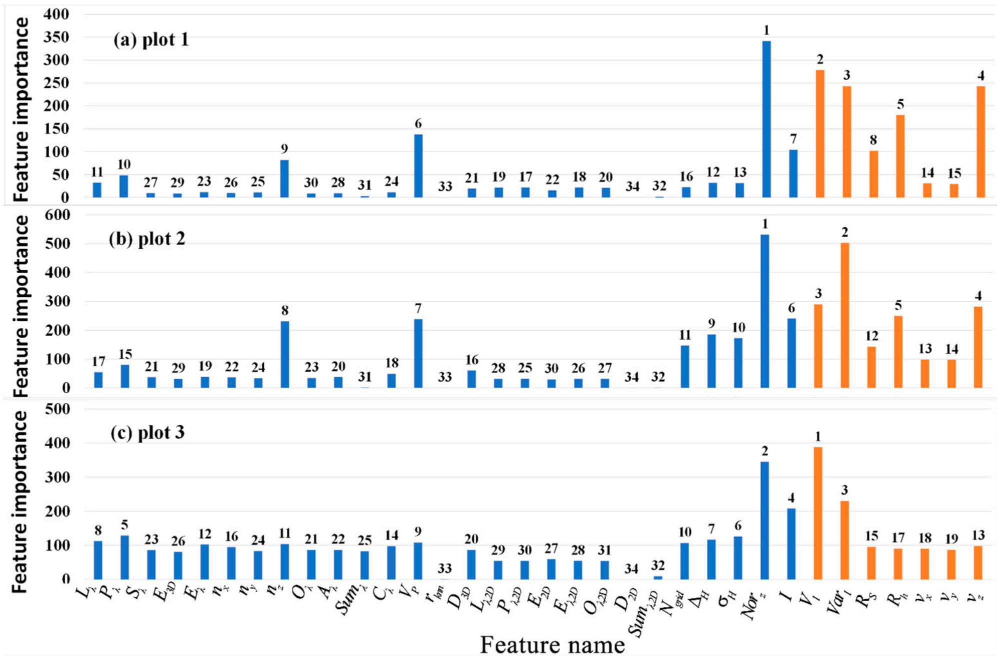


Figure 11. The importance of each feature in different plots (blue: basic features; orange: voxel-based features). The number on each column represents the ranking.

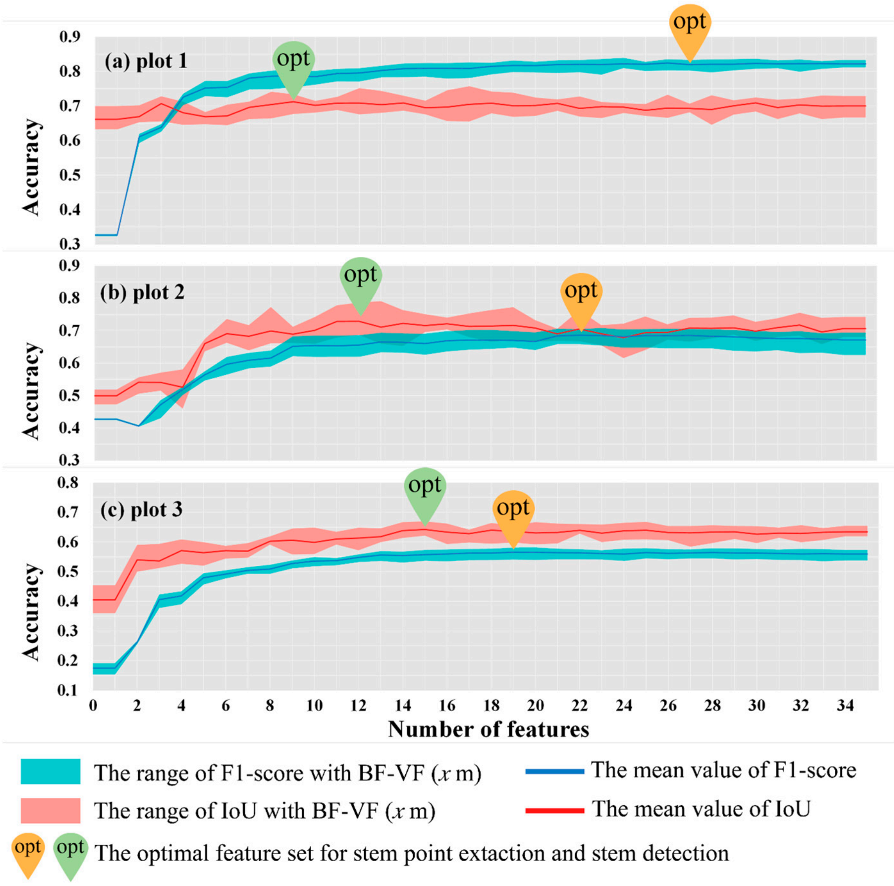


Figure 12. The accuracy of stem point detection (F1-score) and stem detection (IoU) in terms of different feature combinations.

A detailed comparison of the two feature selection methods is shown in Table 7. The relation between the point-based and stem-based results in terms of recall (completeness) and precision (correctness) is not absolute either, which is caused by three probable reasons. First, it is important to note that detecting more stem points may not necessarily lead to the detection of more tree stems, as the additionally detected points may belong to the stems that have already been covered (Figure 13a–c). Second, even more non-stem points labeled as stem points can lower the precision significantly, and they can be removed easily because of obviously different characteristics. Third, the addition of features enhances the ability to discriminate details (e.g., branches), but it can lead to adjacent tree stems being unable to separate (Figure 13d–f). Thus, the key to stem detection may be preserving points that can cover stem objects as more as possible. In summary, compared with the point-based feature selection method, the proposed feature selection method can improve the accuracy of stem detection and reduce the feature dimension. In addition, due to distinct optimal feature sets containing different features and numbers, the universality of features is restricted. To address this issue, the union of optimal feature sets from the three plots is regarded as the final feature set F_{final}, consisting of 16 features (Figure 14), and representing the most effective components of optimal feature combinations in each plot. The results are presented in Table 8. Despite a slight decrease in the stem detection accuracy, it still surpasses the results obtained using the point-based method, thereby validating the portability of this feature set. Finally, Figure 15 illustrates the stem detection result using the proposed feature selection method in each plot, with the slope of different plots and the distance to the scanner position. It can be seen that most undetected trees are distributed far away from the scanner as the stems at a large distance are prone to be obscured by the occlusion of the foreground objects.

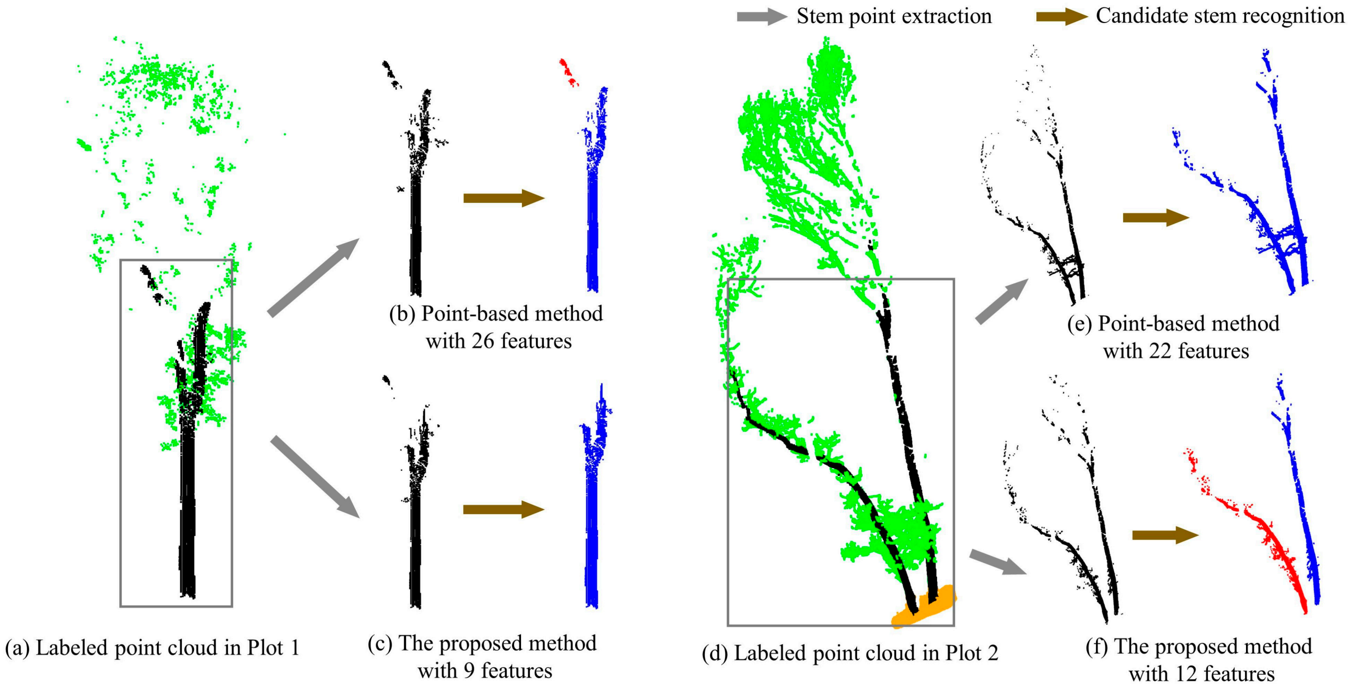


Figure 13. Comparison of point-based methods and proposed methods in stem point extraction and candidate stem recognition. Black: stem points; green: foliage points; orange: ground points; blue and red: different candidate stems.

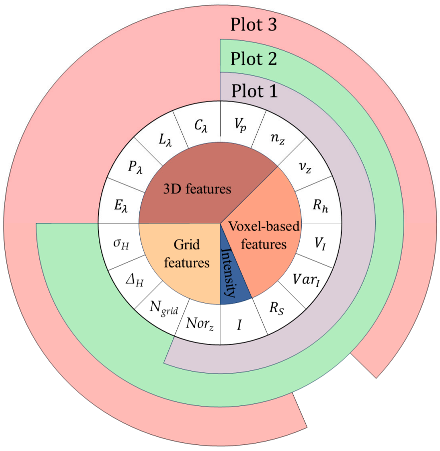


Figure 14. The final feature set composed of optimal feature combinations from different plots.

Table 7. The accuracies of stem point extraction and stem detection for different optimal feature combination in three plots.

Plot ID	Feature Sets	Feature Number	Stem Point Extraction (%)			Stem Detection (%)		
			Precision	Recall	F1-Score	Correctness	Completeness	IoU
1	Proposed method	9	69.01	91.68	78.72	80.91	85.61	**71.23**
	Point-based method	26	74.66	92.06	**82.44**	77.96	86.31	69.39
2	Proposed method	12	57.28	76.88	65.55	81.51	87.45	**72.86**
	Point-based method	22	65.29	72.37	**68.57**	75.58	91.30	70.50
3	Proposed method	15	50.08	62.79	55.71	74.14	82.75	**64.20**
	Point-based method	19	51.21	63.20	**56.56**	73.53	82.37	63.55

Table 8. The accuracies of stem point extraction and stem detection for final feature set F_{final} in three plots.

Plot ID	Feature Number	Stem Point Extraction (%)			Stem Detection (%)		
		Precision	Recall	F1-Score	Correctness	Completeness	IoU
1	16 (F_{final})	70.19	91.79	79.53	80.20	85.96	70.85
2		61.89	73.20	66.97	77.90	90.01	71.61
3		52.08	61.38	56.33	74.04	82.40	63.94

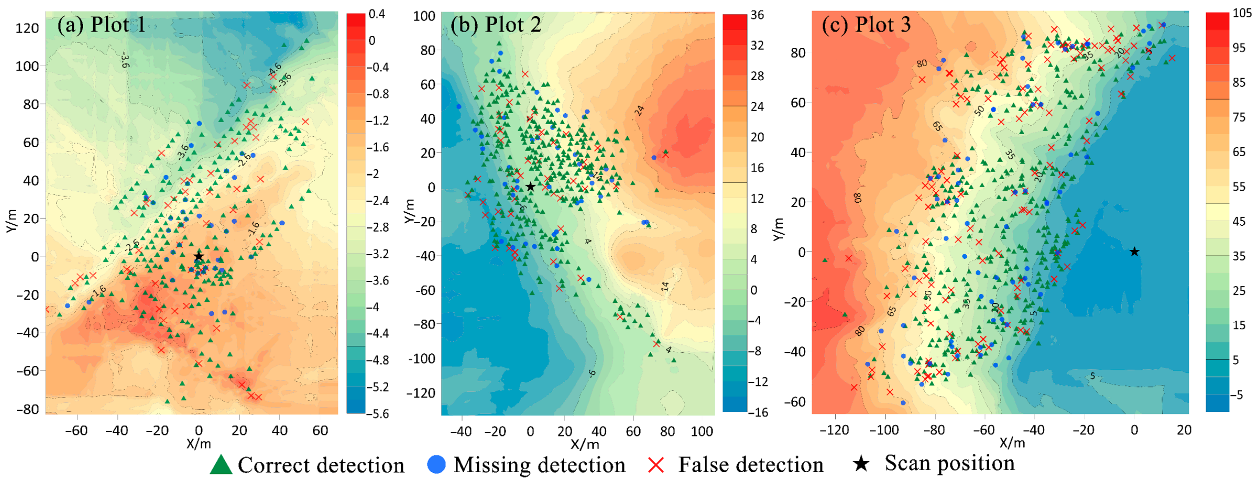


Figure 15. Stem maps from the position calculation. The altitude value is rendered using the color legend on the right.

4.3. Comparison with Feature-Based Methods

The final feature set obtained by using the feature selection method in this paper is compared with four feature combinations used in previous studies [28,31,33,38], as shown in Table 9. The experimental data were collected from different scans in the three research areas outlined in Section 2 to verify the effectiveness of the proposed method in the plot with similar structural distributions and stem characteristics.

Table 9. Comparison of different feature combinations.

Plot ID	Feature Sets	Feature Number	Stem Point Extraction (%)			Stem Detection (%)		
			Precision	Recall	F1-Score	Correctness	Completeness	IoU
1	Proposed method	16	81.14	80.97	81.05	90.00	**91.53**	**83.08**
	Featured in [28]	9	80.13	78.00	79.05	90.91	88.98	81.71
	Featured in [31]	30	**83.28**	**83.82**	**83.55**	**90.00**	**91.53**	**83.08**
	Featured in [33]	6	62.10	65.84	63.92	84.91	76.27	67.16
	Featured in [38]	23	81.27	78.86	80.05	88.98	92.37	82.89
2	Proposed method	16	66.47	89.09	76.14	81.45	86.01	**71.91**
	Featured in [28]	9	65.59	88.96	75.51	78.16	80.15	65.49
	Featured in [31]	30	**70.96**	89.59	**79.19**	80.95	**86.51**	71.88
	Featured in [33]	6	53.26	75.94	62.61	78.68	63.87	54.45
	Featured in [38]	23	69.07	**91.91**	78.87	**82.27**	84.99	71.83
3	Proposed method	16	50.88	61.94	55.87	76.18	**84.10**	**66.59**
	Featured in [28]	9	47.76	60.13	53.24	78.39	74.31	61.68
	Featured in [31]	30	**61.48**	**76.53**	**68.18**	77.43	82.87	66.75
	Featured in [33]	6	42.58	62.79	50.75	**82.72**	61.47	54.47
	Featured in [38]	23	47.84	59.83	53.17	76.55	82.87	66.10

Compared to the four feature combinations in previous studies, the feature number selected using the proposed method and the accuracy of the stem point extraction reach middle to upper levels, while the accuracy of stem detection is the highest in plots 1 and 2, and only slightly lower than that of [31] in plot 3. First, 30 features were calculated at five scales for stem–leaf separation in [31], and the highest accuracy of stem point extraction was obtained, particularly in plot 3. However, the stem detection accuracy did not improve significantly, which aligned with the conclusions of [21] and this paper; although most stem points are extracted, stem detection accuracy is still influenced by the scene's complexity. Secondly, the feature sets used in [28,33] resulted in lower accuracies of stem point extraction and stem detection compared with our method. However, it is worth noting that the feature set in [38] produced a stem detection accuracy that was nearly identical to that obtained by the feature set selected in this paper. This is possibly due to the use of voxel-based features, which are advantageous for stem detection. Additionally, even though the stem detection accuracy of the feature sets used in [31,38] is similar to this paper, they use more features that require more processing time and memory. Therefore, the stem-based feature selection method has certain advantages, which can ensure the accuracy of stem detection and avoid the effective feature loss and the effect of redundant features.

4.4. Experiments on Benchmark Datasets

To further validate the effectiveness of the proposed method in stem detection, we conducted verification using the second dataset, the benchmark dataset. We selected points from the multi-scan TLS datasets at a regular interval of 0.01 m to reduce the number of points and calculated each feature of the final feature set F_{final}. Before computing the voxel features, a voxel size of 0.6 m was set. Tables 10 and 11 show the accuracy of stem detection of the six plots using both single- and multi-scan TLS datasets. The mean accuracy measured in there is more suitable for comparison with other methods.

Table 10. The stem detection accuracy using single-scan TLS datasets.

Plot ID		1	2	3	4	5	6	Mean
Plot Complexity		**Easy**	**Easy**	**Medium**	**Medium**	**Difficult**	**Difficult**	
Completeness (%)	Zhang et al. [21]	80.39	57.14	46.62	34.62	13.74	8.47	40.16
	Chang et al. [56]	88.20	63.10	52.00	44.90	22.90	14.40	47.58
	Proposed method	**88.24**	**77.38**	**68.24**	**70.51**	**59.54**	**29.24**	**66.12**
Correctness (%)	Zhang et al. [21]	**97.62**	92.31	**100.00**	**100.00**	**90.00**	**100.00**	**96.66**
	Chang et al. [56]	95.70	**94.60**	95.10	94.60	61.00	50.70	81.95
	Proposed method	91.84	87.84	94.39	80.88	81.28	95.83	90.89
Mean Accuracy (%)	Zhang et al. [21]	88.17	70.59	63.59	51.43	23.84	15.63	52.21
	Chang et al. [56]	**90.90**	75.70	68.40	60.90	33.30	22.40	58.60
	Proposed method	90.00	**82.28**	**79.22**	**75.34**	**68.72**	**44.81**	**73.40**

Table 11. The stem detection accuracy using multi-scan TLS datasets.

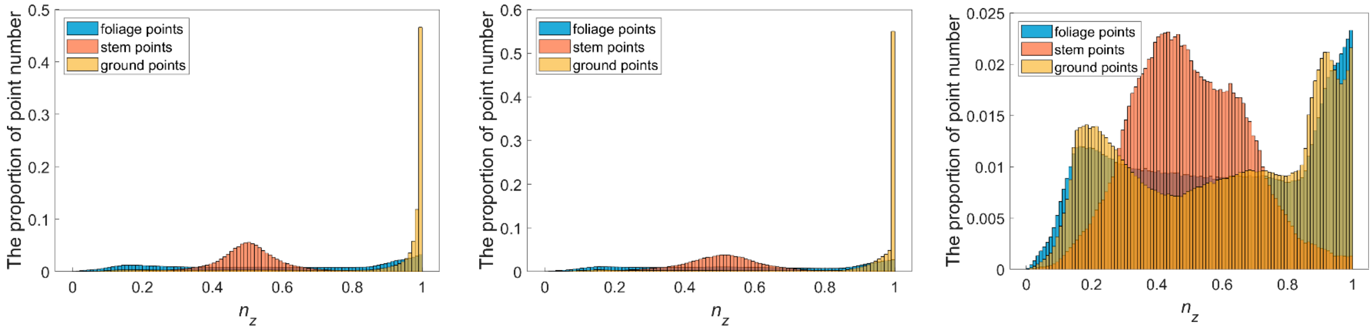
Plot ID		1	2	3	4	5	6	Mean
Plot Complexity		**Easy**	**Easy**	**Medium**	**Medium**	**Difficult**	**Difficult**	
Completeness (%)	Wang et al. [2]	90.20	**92.90**	**79.10**	**78.20**	64.10	52.50	76.17
	Zhang et al. [21]	86.27	82.14	61.49	57.69	45.80	26.27	59.94
	Chang et al. [56]	94.10	94.00	78.40	74.30	64.80	52.70	**76.38**
	Proposed method	**95.83**	89.29	72.97	69.23	**69.47**	**59.32**	76.02
Correctness (%)	Wang et al. [2]	95.80	81.30	72.20	70.10	67.20	70.90	76.25
	Zhang et al. [21]	**97.78**	**95.83**	**100.00**	**97.83**	**93.75**	**98.41**	**97.27**
	Chang et al. [56]	94.10	89.70	94.30	93.50	85.80	81.60	89.83
	Proposed method	93.88	91.46	87.80	81.82	79.13	89.17	87.21
Mean Accuracy (%)	Wang et al. [2]	92.90	86.70	75.50	74.00	65.60	60.30	75.83
	Zhang et al. [21]	91.67	88.46	76.15	72.58	61.54	41.47	71.98
	Chang et al. [56]	94.10	**91.90**	**85.60**	**82.80**	73.90	64.10	**82.07**
	Proposed method	**94.85**	90.36	79.70	75.00	**73.98**	**71.25**	80.86

From the stem detection results of the single-scan scenarios, the proposed method achieves a superior balance between completeness and correctness, while also attaining the highest mean accuracy. Although correctness reached 100% in plots 2, 3, and 6 in [21], it was achieved at the cost of reduced completeness, indicating that a considerable number of stems in the plots remained undiscovered. Similarly, in the multi-scan scenarios, the proposed method exhibits lower correctness compared to [21] but offers higher completeness. The mean accuracy of the proposed method across all plots is slightly lower than that achieved using the deep learning approaches [56]. However, it is worth noting that, whether in single- or multi-scan mode, the proposed method detects more trees in difficult plots, and the trade-off in correctness remains acceptable.

5. Discussion

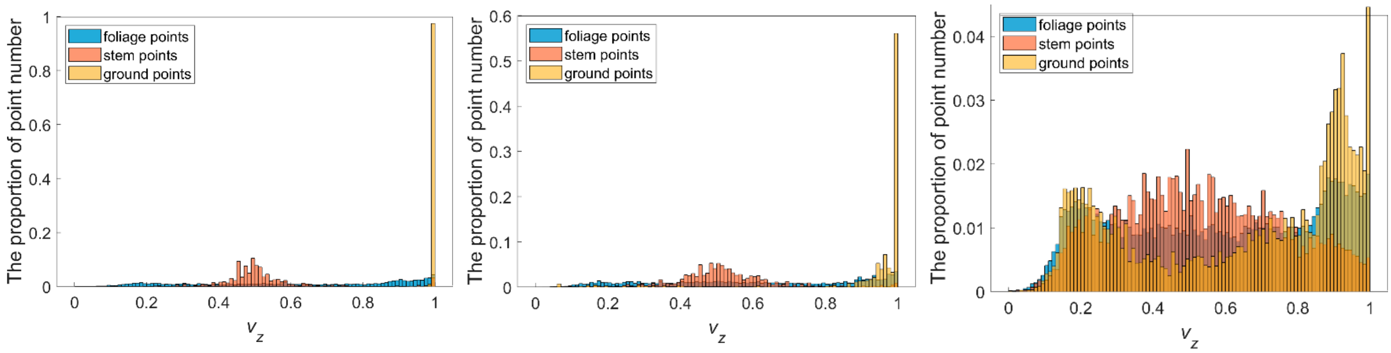
In previous studies, feature combinations are commonly evaluated from the view of point-based results [40,47]. Here, we focus on stem-based results, as applications of the point cloud under a forest scene are usually based on stem instance, e.g., stem mapping [3], DBH estimation [10], and biomass estimation [57]. For stem detection, we only require partial stem points to fit a stem model (e.g., cylinder); thus, missing some stem points may not cause stem misdetection. On the other side, as non-stem points can be filtered via clustering [25] or cylindrical fitting [12] in the process of stem detection, the points falsely labeled as stem in classification may not cause a false detection. This also explains the above experimental results that the improving stem point extraction result may not improve the stem detection result. Considering the above points, we try to construct feature combination from both point-based and stem-based evaluations. Although feature combination is not directly related to stem-based results, an obvious correlation between them can be observed (Figure 12). The comparison of different feature combinations also shows that the stem-based feature selection method is effective. Compared with point-based feature selection, less features are used to obtain similar or better stem-based results in our method, which significantly reduces the computational burden with respect to processing time and memory consumption. Moreover, compared to constructing feature combination using prior knowledge, the feature set selected in this study performed well and struck a good balance between stem detection in various forest scenes and feature redundancy. In terms of the scanning mode, we first tested and validated the effectiveness of the proposed method on single-scan TLS data. The single-scan mode serves as the foundation for the multi-scan mode, and the stem positions obtained from the single-scan mode can be used for TLS point cloud registration [58–60] and even assist in the fusion of multi-sensor data [61,62]. Subsequent testing on the benchmark dataset indicated the advantages of the proposed method in all plots under the single-scan mode and in two different plots under the multi-scan mode. However, the difference in accuracy is related to the plot size and forest stand complexity [63,64]. In regions with dense vegetation within the plots and along the plot edges, the stem detection accuracy significantly decreases.

The feature set obtained using the stem-based selection method in this study demonstrates the significant contributions of voxel-based features. Compared to basic features, the inclusion of voxel-based features shows an improved accuracy in stem point extraction across different voxel sizes (Figure 10a). However, for stem detection, the optimal results were achieved in three different plots with voxel sizes of 1.4 m, 0.8 m, and 0.6 m (Figure 10b), indicating that the determination of the voxel size should consider both the stem density and understory vegetation richness. Using excessively large voxel sizes may lead to the covering of stem points and non-stem points (e.g., foliage and shrub) within a single voxel, while overly small voxel sizes may not be able to contain enough stem points, especially in regions with significant density variations. Both situations can potentially result in errors in stem detection caused by a decrease in the capability of feature expression. The features I, V_I, and Var_I related to intensity information and the features Δ_H, σ_H, and Nor_z related to height are included in the optimal feature set and make significant contributions to all three plots. This is because the ground, stem, and foliage have different materials and spatial distributions. Moreover, the normal vector related with point distribution in the vertical direction was often used in previous studies [3,24,28], as it is consistent with the law that trees grow upward. But normal vector-based features (n_z and v_z) did not show a high importance across all plots in our test. Figures 16 and 17 show the histogram of normal vector-based features, n_z and v_z, respectively. In plots 1 and 2, the n_z and v_z of stems are distributed in significantly different intervals from the other two types of objects, while the normal vector distribution of three objects is relatively divergent in plot 3, and there is a similar point number of three objects under the same interval (Figures 16 and 17c), which leads to a decrease in the effectiveness of normal vector-based features. The possible reasons lie in that the trees start branching from a height that is close to the ground, and the terrain slope close to 42° also affects the growth direction in plot 3, which is similar to the conclusion in [27]. Therefore, the performance of normal vector-based features is reduced in plot 3. Moreover, the features Sum_λ, D_{3D}, $Sum_{\lambda,2D}$, and D_{2D} are always among the least relevant features, which is consistent to similar studies involving the features [32].



(**a**) plot 1 (**b**) plot 2 (**c**) plot 3

Figure 16. The histogram of the normal vectors' z-values n_z of foliage points stem points and ground points in different plots.



(**a**) plot 1 (**b**) plot 2 (**c**) plot 3

Figure 17. The histogram of the normal vectors' z-values v_z of foliage points stem points and ground points in different plots.

6. Conclusions

In this study, we focus on the improvement in stem detection results by proposing seven voxel-based features and optimizing the feature combination using a stem-based feature selection method. The result shows that the voxel-based features are beneficial for stem point extraction and stem detection in forest scenes. Moreover, our study highlights the importance of tailoring feature combinations specifically for stem detection tasks, as the benefits observed in point-based results may not always extend to stem-based results. The proposed method can provide more reliable feature combination for stem detection in different forests. Compared with the feature set in previous studies, the feature combination constructed using our method achieves a good balance between the feature number and accuracy. Regarding stem detection, the proposed method also yields comparable results on the benchmark dataset and exhibits better mean accuracy on difficult plots. It is also found that the performance of some features related with point distribution in the vertical direction is significantly affected by the terrain slopes and tree characteristics. In general, the accuracy of stem detection will decrease with an increasing forest complexity, because complex forests usually have a strong and increasing likelihood of occlusion. Integrating data from multiple sensors (e.g., TLS, UAV, and ALS) has the potential to enhance the point cloud quality in complex forests, improving stem detection accuracy and enabling a more precise estimation of forest structural parameters. Additionally, the stem positions calculated in this paper can be valuable for assisting in point cloud registration or fusion. Nevertheless, this requires further investigation and study.

Author Contributions: Conceptualization, M.C.; data curation, M.C.; formal analysis, X.L.; funding acquisition, M.C.; investigation, F.M. and L.Z.; methodology, X.L.; project administration, J.P.; resources, M.C.; supervision, M.C., J.P. and L.Z.; validation, X.L.; visualization, X.L. and F.M.;

writing—original draft, X.L.; writing—review and editing, M.C. and L.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the Key R&D Program of Ningxia Autonomous Region (grant no. 2022CMG02014), the National Natural Science Foundation of China (grant no. 41801394), and the Open Fund of Key Laboratory of Urban Resources Monitoring and Simulation, Ministry of Natural Resources (grant no. KF-2021-06-102).

Data Availability Statement: Not applicable.

Acknowledgments: The authors would like to thank Zhang (Xinyi Zhang) for her constructive comments and suggestions that improved the quality of this work.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Wilkes, P.; Lau, A.; Disney, M.; Calders, K.; Burt, A.; Gonzalez de Tanago, J.; Bartholomeus, H.; Brede, B.; Herold, M. Data Acquisition Considerations for Terrestrial Laser Scanning of Forest Plots. *Remote Sens. Environ.* **2017**, *196*, 140–153. [CrossRef]
2. Wang, D. Unsupervised Semantic and Instance Segmentation of Forest Point Clouds. *ISPRS J. Photogramm. Remote Sens.* **2020**, *165*, 86–97. [CrossRef]
3. Liang, X.; Litkey, P.; Hyyppa, J.; Kaartinen, H.; Vastaranta, M.; Holopainen, M. Automatic Stem Mapping Using Single-Scan Terrestrial Laser Scanning. *IEEE Trans. Geosci. Remote Sens.* **2012**, *50*, 661–670. [CrossRef]
4. Ye, W.; Qian, C.; Tang, J.; Liu, H.; Fan, X.; Liang, X.; Zhang, H. Improved 3D Stem Mapping Method and Elliptic Hypothesis-Based DBH Estimation from Terrestrial Laser Scanning Data. *Remote Sens.* **2020**, *12*, 352. [CrossRef]
5. Cabo, C.; Ordóñez, C.; López-Sánchez, C.A.; Armesto, J. Automatic Dendrometry: Tree Detection, Tree Height and Diameter Estimation Using Terrestrial Laser Scanning. *Int. J. Appl. Earth Obs. Geoinf.* **2018**, *69*, 164–174. [CrossRef]
6. Pitkänen, T.P.; Raumonen, P.; Kangas, A. Measuring Stem Diameters with TLS in Boreal Forests by Complementary Fitting Procedure. *ISPRS J. Photogramm. Remote Sens.* **2019**, *147*, 294–306. [CrossRef]
7. Rouzbeh Kargar, A.; MacKenzie, R.A.; Apwong, M.; Hughes, E.; van Aardt, J. Stem and Root Assessment in Mangrove Forests Using a Low-Cost, Rapid-Scan Terrestrial Laser Scanner. *Wetl. Ecol. Manag.* **2020**, *28*, 883–900. [CrossRef]
8. Pueschel, P.; Newnham, G.; Rock, G.; Udelhoven, T.; Werner, W.; Hill, J. The Influence of Scan Mode and Circle Fitting on Tree Stem Detection, Stem Diameter and Volume Extraction from Terrestrial Laser Scans. *ISPRS J. Photogramm. Remote Sens.* **2013**, *77*, 44–56. [CrossRef]
9. Koreň, M.; Mokroš, M.; Bucha, T. Accuracy of Tree Diameter Estimation from Terrestrial Laser Scanning by Circle-Fitting Methods. *Int. J. Appl. Earth Obs. Geoinf.* **2017**, *63*, 122–128. [CrossRef]
10. Maas, H.-G.; Bienert, A.; Scheller, S.; Keane, E. Automatic Forest Inventory Parameter Determination from Terrestrial Laser Scanner Data. *Int. J. Remote Sens.* **2008**, *29*, 1579–1593. [CrossRef]
11. Bu, G.; Wang, P. Adaptive Circle-Ellipse Fitting Method for Estimating Tree Diameter Based on Single Terrestrial Laser Scanning. *J. Appl. Remote Sens.* **2016**, *10*, 026040. [CrossRef]
12. Liang, X.; Kankare, V.; Yu, X.; Hyyppa, J.; Holopainen, M. Automated Stem Curve Measurement Using Terrestrial Laser Scanning. *IEEE Trans. Geosci. Remote Sens.* **2014**, *52*, 1739–1748. [CrossRef]
13. Liu, G.; Wang, J.; Dong, P.; Chen, Y.; Liu, Z. Estimating Individual Tree Height and Diameter at Breast Height (DBH) from Terrestrial Laser Scanning (TLS) Data at Plot Level. *Forests* **2018**, *9*, 398. [CrossRef]
14. Haala, N.; Reulke, R.; Thies, M.; Aschoff, T. Combination of terrestrial Laser Scanning with high resolution panoramic Images for Investigations in Forest Applications and tree species recognition. In Proceedings of the Panoramic Photogrammetry Workshop, Berlin, Germany, 24–25 February 2005.
15. Forsman, P.; Halme, A. 3-D mapping of natural environments with trees by means of mobile perception. *IEEE Trans. Robot.* **2005**, *21*, 482–490. [CrossRef]
16. Forsman, M.; Börlin, N.; Holmgren, J. Estimation of tree stem attributes using terrestrial photogrammetry. In Proceedings of the ISPRS-International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Melbourne, Australia, 25 August–1 September 2012; pp. 261–265.
17. Xia, S.; Chen, D.; Peethambaran, J.; Wang, P.; Xu, S. Point Cloud Inversion: A Novel Approach for the Localization of Trees in Forests from TLS Data. *Remote Sens.* **2021**, *13*, 338. [CrossRef]
18. Huo, L.; Lindberg, E.; Holmgren, J. Towards low vegetation identification: A new method for tree crown segmentation from LiDAR data based on a symmetrical structure detection algorithm (SSD). *Remote Sens. Environ.* **2022**, *270*, 112857. [CrossRef]
19. Béland, M.; Widlowski, J.-L.; Fournier, R.A.; Côté, J.-F.; Verstraete, M.M. Estimating Leaf Area Distribution in Savanna Trees from Terrestrial LiDAR Measurements. *Agric. For. Meteorol.* **2011**, *151*, 1252–1266. [CrossRef]
20. Ma, L.; Zheng, G.; Eitel, J.U.H.; Moskal, L.M.; He, W.; Huang, H. Improved Salient Feature-Based Approach for Automatically Separating Photosynthetic and Nonphotosynthetic Components Within Terrestrial Lidar Point Cloud Data of Forest Canopies. *IEEE Trans. Geosci. Remote Sens.* **2016**, *54*, 679–696. [CrossRef]

21. Zhang, W.; Wan, P.; Wang, T.; Cai, S.; Chen, Y.; Jin, X.; Yan, G. A Novel Approach for the Detection of Standing Tree Stems from Plot-Level Terrestrial Laser Scanning Data. *Remote Sens.* **2019**, *11*, 211. [CrossRef]
22. Wang, D.; Momo Takoudjou, S.; Casella, E. LeWoS: A Universal Leaf-wood Classification Method to Facilitate the 3D Modelling of Large Tropical Trees Using Terrestrial LiDAR. *Methods Ecol. Evol.* **2020**, *11*, 376–389. [CrossRef]
23. Li, S.; Dai, L.; Wang, H.; Wang, Y.; He, Z.; Lin, S. Estimating Leaf Area Density of Individual Trees Using the Point Cloud Segmentation of Terrestrial LiDAR Data and a Voxel-Based Model. *Remote Sens.* **2017**, *9*, 1202. [CrossRef]
24. Wang, D.; Hollaus, M.; Puttonen, E.; Pfeifer, N. Automatic and Self-Adaptive Stem Reconstruction in Landslide-Affected Forests. *Remote Sens.* **2016**, *8*, 974. [CrossRef]
25. Xia, S.; Wang, C.; Pan, F.; Xi, X.; Zeng, H.; Liu, H. Detecting Stems in Dense and Homogeneous Forest Using Single-Scan TLS. *Forests* **2015**, *6*, 3923–3945. [CrossRef]
26. Olofsson, K.; Holmgren, J. Single Tree Stem Profile Detection Using Terrestrial Laser Scanner Data, Flatness Saliency Features and Curvature Properties. *Forests* **2016**, *7*, 207. [CrossRef]
27. Zhu, X.; Skidmore, A.K.; Darvishzadeh, R.; Niemann, K.O.; Liu, J.; Shi, Y.; Wang, T. Foliar and Woody Materials Discriminated Using Terrestrial LiDAR in a Mixed Natural Forest. *Int. J. Appl. Earth Obs. Geoinf.* **2018**, *64*, 43–50. [CrossRef]
28. Chen, M.; Wan, Y.; Wang, M.; Xu, J. Automatic Stem Detection in Terrestrial Laser Scanning Data with Distance-Adaptive Search Radius. *IEEE Trans. Geosci. Remote Sens.* **2018**, *56*, 2968–2979. [CrossRef]
29. Hackel, T.; Wegner, J.D.; Schindler, K. Fast Semantic Segmentation of 3D Point Clouds with Strongly Varying Density. *ISPRS Ann. Photogramm. Remote Sens. Spat. Inf. Sci.* **2016**, *3*, 177–184. [CrossRef]
30. Becker, C.; Rosinskaya, E.; Häni, N.; D'Angelo, E.; Strecha, C. Classification of Aerial Photogrammetric 3D Point Clouds. *Photogramm. Eng. Remote Sens.* **2018**, *84*, 287–295. [CrossRef]
31. Krishna Moorthy, S.M.; Calders, K.; Vicari, M.B.; Verbeeck, H. Improved Supervised Learning-Based Approach for Leaf and Wood Classification from LiDAR Point Clouds of Forests. *IEEE Trans. Geosci. Remote Sens.* **2020**, *58*, 3057–3070. [CrossRef]
32. Weinmann, M.; Jutzi, B.; Mallet, C. Feature Relevance Assessment for the Semantic Interpretation of 3D Point Cloud Data. *ISPRS Ann. Photogramm. Remote Sens. Spat. Inf. Sci.* **2013**, *II5*, 313–318. [CrossRef]
33. Vicari, M.B.; Disney, M.; Wilkes, P.; Burt, A.; Calders, K.; Woodgate, W. Leaf and Wood Classification Framework for Terrestrial LiDAR Point Clouds. *Methods Ecol. Evol.* **2019**, *10*, 680–694. [CrossRef]
34. Lim, E.H.; Suter, D. 3D Terrestrial LIDAR Classifications with Super-Voxels and Multi-Scale Conditional Random Fields. *Comput.-Aided Des.* **2009**, *41*, 701–710. [CrossRef]
35. Plaza-Leiva, V.; Gomez-Ruiz, J.A.; Mandow, A.; García-Cerezo, A. Voxel-Based Neighborhood for Spatial Shape Pattern Classification of Lidar Point Clouds with Supervised Learning. *Sensors* **2017**, *17*, 594. [CrossRef] [PubMed]
36. Aijazi, A.; Checchin, P.; Trassoudaine, L. Segmentation Based Classification of 3D Urban Point Clouds: A Super-Voxel Based Approach with Evaluation. *Remote Sens.* **2013**, *5*, 1624–1650. [CrossRef]
37. Xu, Y.; Tong, X.; Stilla, U. Voxel-Based Representation of 3D Point Clouds: Methods, Applications, and Its Potential Use in the Construction Industry. *Autom. Constr.* **2021**, *126*, 103675. [CrossRef]
38. Xu, S.; Zhou, K.; Sun, Y.; Yun, T. Separation of Wood and Foliage for Trees from Ground Point Clouds Using a Novel Least-Cost Path Model. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2021**, *14*, 6414–6425. [CrossRef]
39. Brolly, G.; Király, G.; Lehtomäki, M.; Liang, X. Voxel-Based Automatic Tree Detection and Parameter Retrieval from Terrestrial Laser Scans for Plot-Wise Forest Inventory. *Remote Sens.* **2021**, *13*, 542. [CrossRef]
40. Weinmann, M.; Jutzi, B.; Mallet, C.; Weinmann, M. Geometric Features and Their Relevance for 3D Point Cloud Classification. *ISPRS Ann. Photogramm. Remote Sens. Spat. Inf. Sci.* **2017**, *4*, 157–164. [CrossRef]
41. Weinmann, M.; Jutzi, B.; Hinz, S.; Mallet, C. Semantic Point Cloud Interpretation Based on Optimal Neighborhoods, Relevant Features and Efficient Classifiers. *ISPRS J. Photogramm. Remote Sens.* **2015**, *105*, 286–304. [CrossRef]
42. Li, Q.; Cheng, X. Comparison of Different Feature Sets for TLS Point Cloud Classification. *Sensors* **2018**, *18*, 4206. [CrossRef]
43. Liang, X.; Hyyppä, J.; Kaartinen, H.; Lehtomäki, M.; Pyörälä, J.; Pfeifer, N.; Holopainen, M.; Brolly, G.; Francesco, P.; Hackenberg, J.; et al. International Benchmarking of Terrestrial Laser Scanning Approaches for Forest Inventories. *ISPRS J. Photogramm. Remote Sens.* **2018**, *144*, 137–179. [CrossRef]
44. Farabet, C.; Couprie, C.; Najman, L.; LeCun, Y. Learning Hierarchical Features for Scene Labeling. *IEEE Trans. Pattern Anal. Mach. Intell.* **2012**, *35*, 1915–1929. [CrossRef] [PubMed]
45. Weinmann, M.; Mallet, C.; Jutzi, B. Involving Different Neighborhood Types for the Analysis of Low-Level Geometric 2D and 3D Features and Their Relevance for Point Cloud Classification. *Publ. Dtsch. Ges. Für. Photogramm.* **2017**, *26*, 179–191.
46. Atik, M.E.; Duran, Z.; Seker, D.Z. Machine Learning-Based Supervised Classification of Point Clouds Using Multiscale Geometric Features. *ISPRS Int. J. Geo-Inf.* **2021**, *10*, 187. [CrossRef]
47. Günen, M.A. Adaptive Neighborhood Size and Effective Geometric Features Selection for 3D Scattered Point Cloud Classification. *Appl. Soft Comput.* **2022**, *115*, 108196. [CrossRef]
48. Breiman, L. Random Forests. *Mach. Learn.* **2001**, *45*, 5–32. [CrossRef]
49. Weinmann, M.; Urban, S.; Hinz, S.; Jutzi, B.; Mallet, C. Distinctive 2D and 3D features for automated large-scale scene analysis in urban areas. *Comput. Graph.* **2015**, *49*, 47–57. [CrossRef]
50. Lalonde, J.-F.; Vandapel, N.; Huber, D.F.; Hebert, M. Natural Terrain Classification Using Three-Dimensional Ladar Data for Ground Robot Mobility. *J. Field Robot.* **2006**, *23*, 839–861. [CrossRef]

51. Wang, D.; Hollaus, M.; Pfeifer, N. Feasibility of Machine Learning Methods for Separating Wood and Leaf Points from Terrestrial Laser Scanning Data. *ISPRS Ann. Photogramm. Remote Sens. Spat. Inf. Sci.* **2017**, *42*, 157–164. [CrossRef]
52. Gregorutti, B.; Michel, B.; Saint-Pierre, P. Correlation and variable importance in random forests. *Stat. Comput.* **2017**, *27*, 659–678. [CrossRef]
53. Olofsson, K.; Holmgren, J.; Olsson, H. Tree Stem and Height Measurements Using Terrestrial Laser Scanning and the RANSAC Algorithm. *Remote Sens.* **2014**, *6*, 4323–4344. [CrossRef]
54. Niemeyer, J.; Rottensteiner, F.; Soergel, U. Contextual classification of lidar data and building object detection in urban areas. *ISPRS J. Photogramm. Remote Sens.* **2014**, *87*, 152–165. [CrossRef]
55. Chen, M.; Pan, J.; Xu, J. Classification of terrestrial laser scanning data with density-adaptive geometric features. *IEEE Geosci. Remote Sens. Lett.* **2018**, *15*, 1795–1799. [CrossRef]
56. Chang, L.; Fan, H.; Zhu, N.; Dong, Z. A Two-stage Approach for Individual Tree Segmentation from TLS Point Clouds. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2022**, *15*, 8682–8693. [CrossRef]
57. Tao, S.; Guo, Q.; Xu, S.; Su, Y.; Li, Y.; Wu, F. A Geometric Method for Wood-Leaf Separation Using Terrestrial and Simulated Lidar Data. *Photogramm. Eng. Remote Sens.* **2015**, *81*, 767–776. [CrossRef]
58. Liang, X.; Hyyppä, J. Automatic stem mapping by merging several terrestrial laser scans at the feature and decision levels. *Sensors* **2013**, *13*, 1614–1634. [CrossRef]
59. Tremblay, J.F.; Béland, M. Towards operational marker-free registration of terrestrial lidar data in forests. *ISPRS J. Photogramm. Remote Sens.* **2018**, *146*, 430–435. [CrossRef]
60. Kelbe, D.; Van Aardt, J.; Romanczyk, P.; Van Leeuwen, M.; Cawse-Nicholson, K. Marker-free registration of forest terrestrial laser scanner data pairs with embedded confidence metrics. *IEEE Trans. Geosci. Remote Sens.* **2016**, *54*, 4314–4330. [CrossRef]
61. Hauglin, M.; Lien, V.; Næsset, E.; Gobakken, T. Geo-referencing forest field plots by co-registration of terrestrial and airborne laser scanning data. *Int. J. Remote Sens.* **2014**, *35*, 3135–3149. [CrossRef]
62. Polewski, P.; Erickson, A.; Yao, W.; Coops, N.; Krzystek, P.; Stilla, U. Object-based coregistration of terrestrial photogrammetric and als point clouds in forested areas. *ISPRS Ann. Photogramm. Remote Sens. Spat. Inf. Sci.* **2016**, *3*, 347–354. [CrossRef]
63. Heinzel, J.; Huber, M. Detecting Tree Stems from Volumetric TLS Data in Forest Environments with Rich Understory. *Remote Sens.* **2016**, *9*, 9. [CrossRef]
64. Pitkänen, T.P.; Raumonen, P.; Liang, X.; Lehtomäki, M.; Kangas, A. Improving TLS-Based Stem Volume Estimates by Field Measurements. *Comput. Electron. Agric.* **2021**, *180*, 105882. [CrossRef]