remote sensing

MDPI

Article

A Self-Adaptive Optimization Individual Tree Modeling Method for Terrestrial LiDAR Point Clouds

Zhenyang Hui, Zhaochen Cai, Bo Liu *, Dajun Li, Hua Liu and Zhuoxuan Li

Faculty of Geomatics, East China University of Technology, Nanchang 330013, China; huizhenyang2008@ecut.edu.cn (Z.H.); 2021110139@ecut.edu.cn (Z.C.); djli@ecut.edu.cn (D.L.); liuhua@ecut.edu.cn (H.L.); 2020110363@ecut.edu.cn (Z.L.)

* Correspondence: liubo@ecut.edu.cn

Abstract: Individual tree modeling for terrestrial LiDAR point clouds always involves heavy computation burden and low accuracy toward a complex tree structure. To solve these problems, this paper proposed a self-adaptive optimization individual tree modeling method. In this paper, we first proposed a joint neighboring growing method to segment wood points into object primitives. Subsequently, local object primitives were optimized to alleviate the computation burden. To build the topology relation among branches, branches were separated based on spatial connectivity analysis. And then the nodes corresponding to each object primitive were adopted to construct the graph structure of the tree. Furthermore, each object primitive was fitted as a cylinder. To revise the local abnormal cylinder, a self-adaptive optimization method based on the constructed graph structure was proposed. Finally, the constructed tree model was further optimized globally based on prior knowledge. Twenty-nine field datasets obtained from three forest sites were adopted to evaluate the performance of the proposed method. The experimental results show that the proposed method can achieve satisfying individual tree modeling accuracy. The mean volume deviation of the proposed method is 1.427 m^3. In the comparison with two other famous tree modeling methods, the proposed method can achieve the best individual tree modeling result no matter which accuracy indicator is selected.

Keywords: individual tree modeling; terrestrial LiDAR; object primitive; joint neighboring growing; global optimization

check for updates

Citation: Hui, Z.; Cai, Z.; Liu, B.; Li, D.; Liu, H.; Li, Z. A Self-Adaptive Optimization Individual Tree Modeling Method for Terrestrial LiDAR Point Clouds. *Remote Sens.* **2022**, *14*, 2545. https://doi.org/10.3390/rs14112545

Academic Editors: Andrea Hevia and Sandra Buján

Received: 4 May 2022
Accepted: 24 May 2022
Published: 26 May 2022

1. Introduction

Through photosynthesis, forests can effectively offset part of the carbon dioxide emissions [1–3]. Thus, it is of great practical significance to conduct forest inventory and explore the carbon sink capacity of forests to protect the Earth's environment and achieve green development [1,4,5]. To better strengthen the investigation of forest resources, promote the study of tree growth mechanism, forest ecosystem simulation, and analysis of the impact of environmental change, it is urgent to analyze the structure change and growth rule change of trees in the process of growth on a single tree scale [6–8].

Conventional forest dynamic survey is generally dependent on massive and chronic forestry inventory data [9,10]. This kind of forestry resource survey not only always needs huge labor and material resources, but it is also inefficient in data acquisition, which will fail to reflect forestry resource change in the short term [11,12]. As a result, it cannot effectively realize the dynamic monitoring of forest resources. In recent years, with the rapid development of three-dimensional LiDAR, this technology has become a vital method in dynamic survey of forestry resources [13,14]. LiDAR technology is an active remote sensing technology, which can actively emit laser pulses and obtain the three-dimensional coordinate information of the target object [15]. The laser pulse emitted by the LiDAR system is able to penetrate vegetation and accurately depict the spatial structure of the vegetation canopy, making itself gradually become an important data source for analysis of vegetation spatial topology [16]. Currently, LiDAR technology has been widely applied in forestry, ecology, botany, and other related fields due to its continuous improvement of measurement accuracy and sampling rate [17,18].

Compared with LiDAR systems with other platforms, terrestrial LiDAR can obtain denser point cloud data. Meanwhile, since the distance of data scanned by terrestrial laser scanner (TLS) is relatively close, the quality of acquired data is relatively high and data error caused by external influence is small [19]. TLS has gradually become the technology that supports for constructing a three-dimensional tree model with high fidelity due to its characteristics such as accuracy, efficiency, and high density of points, etc. [20–22].

According to different principles, tree modeling methods using TLS can be classified into two categories: segmentation-based and skeleton-based [12]. In the segmentation-based methods, tree points are first segmented as a series of subsections, which can be further fitted as small modeling units, such as cylinders. By connecting these modeling units together, a tree model can be constructed. Raumonen et al. [23] developed a famous individual tree modeling method named TreeQSM. In their method, tree points are first segmented as a sequence of cover sets. Thereafter, the neighboring relation among these cover sets can be determined to obtain tree components. Last, tree components are further segmented for fitting cylinders with different radius and length. Hackenberg et al. [24] segmented the tree points into a subpoint cloud using a varying sphere. The sphere center is located at the skeleton of a tree, while its radius is larger than the underlying branch segment. In each subpoint cloud, the sphere center and fitted circle center can be obtained, which will be set as the start point and end point for building the cylinder. Hackenberg et al. [25] further improved this method by first segmenting the points as stem and branch points. In so doing, higher cylinder fitting accuracy can be acquired.

In the skeleton-based methods, the tree skeleton is generally generated by thinning the raw point clouds. The obtained skeleton generally reflects the geometry of the tree. The final tree model can then be reconstructing by estimating the radius of each branch section [26]. Bucksch and Lindenbergh [27] proposed a skeletonization method for a point cloud named CAMPINO. In their method, the skeleton was extracted based on octree-graphs. Although CAMPINO can extract the tree skeleton structure with highly consistent with the original tree points, its performance heavily relies on the voxel size setting. Bucksch et al. [28] further improved this method and proposed a robust skeleton extraction method called SkelTre. The main strength of SkelTre is that this method is insensitive to varying point density and data gaps within point clouds. Wang et al. [26] acquired the tree skeleton by building a minimum distance spanning tree. The main strength is that this method can achieve a complete tree model even if there are data gaps in the raw point clouds. Delagrange et al. [29] presented a tree reconstruction method named as PypeTree. In PypeTree, the skeletal curves were extracted by computing four distinct graph structures. The reconstruction accuracy of this method was evaluated by comparing the skeleton length. Experimental results show that mean automatic reconstruction error was 5.1%. Du et al. [12] proposed a skeleton extraction method named as Adtree, which was further optimized by Fan et al. [30]. In AdTree, the initial tree skeleton was first built based on the Dijkstra shortest path method. Subsequently, the skeleton was optimized by removing adjacent vertices and edges. Finally, the branch model can be built by fitting a series of cylinders. Fan et al. [30] developed an open tool with an end-user interface for AdTree and named it as AdQSM. All the above mentioned methods can be seen as graph-based skeleton extraction methods since these methods generally need to construct a graph structure, such as a minimum spanning tree.

In addition to the graph-based approaches, there are two other kinds of skeleton extraction methods, namely L-1-Medial-based and Laplacian-based. In the L-1-Medial-based methods, the point sets are represented by the median points. The skeleton line is formed by connecting the median points together [31–33]. This method has strong universality and can directly process the original point cloud data. However, if there is a great amount of noise in the point cloud or a large point cloud missing, it may produce wrong output. To solve these problems, Wei et al. [32] combined the advantages of L-1-Medial-based methods and minimum spanning tree and proposed L1-MST approach. In their method, an optimal model considering both structure and point density was established based on the principle of point balance force on the surface of the object. L1-MST can extract the tree skeleton with high precision. However, the extraction accuracy of stem skeleton is poor in dense leaf area.

In the Laplacian-based methods, the Laplace weight matrix of adjacent point clouds is first constructed, and then a series of skeleton points are obtained by iteratively shrinking from the outside to the inside. Finally, skeleton lines are obtained using the minimum spanning tree method [34]. The results of skeleton line extraction in this kind of method depend on the setting of Laplace operator parameters, and inaccurate parameter setting will lead to wrong extraction results. Su et al. [35] first adopted the constrained Laplacian smoothing method to obtain the basic shape of the skeleton, and then the skeleton model was obtained by progressive sampling and optimization. This method can obtain relatively complete tree model and has certain noise resistance, but it is difficult to construct complex canopy model and accurate model for canopy twigs.

To sum up, for individual tree modeling based on terrestrial LiDAR, there are still some problems that need to be solved.

i. The robustness of the tree modeling is not strong. When encountering complex tree structure, the construction of local branch model is prone to error. As a result, the model accuracy is low.
ii. The computation burden for tree modeling is huge. High time complexity and low modeling efficiency lead to the tree modeling methods are not suitable for large-scale and mass tree point cloud modeling.
iii. The modeling methods lack adaptive modeling ability and cannot conduct adaptive optimization and adjustment for some incorrectly constructed model elements.

To solve these problems, this paper proposed a self-adaptive optimization tree modeling method. In this paper, a joint neighboring growing method was first proposed to segment wood points into object primitives. To reduce the amount of calculation and improve the implementation efficiency of modeling, this paper proposed local object primitive self-adaptive constraint adjustment method. Then, branch topology relation was built using graph structure to strengthen the robustness of modeling. Subsequently, local and global optimization was applied to the built tree model to improve the modeling accuracy. This study is expected to propose a tree modeling method with high accuracy, strong robustness and wide applicability, thereby providing contribution to forest inventory.

2. Method

In this paper, we proposed a self-adaptive three-dimensional tree structure modeling method. The flowchart of the proposed method is shown in Figure 1. The individual tree point clouds are first conducted the wood and leaf separation to obtain the wood points. In this paper, the LeWoS method proposed by Wang et al. [36] was applied for wood and leaf separation. In LeWoS, recursive segmentation was first conducted to the individual tree points. Then, the wood components were separated based on the characteristics of each segment, such as linearity and size. After wood and leaf separation, the wood points were further segmented as object primitives based on the neighboring growing method proposed in this paper. For building the topological relation among the branches, the single branches were also separated from each other according to the spatial connectivity during the object primitive segmentation. Note that in the initial wood points segmentation, the segmented results are generally oversegmented. It is because that only the similar neighboring points can be segmented as one object primitive. Obviously, these object primitives can be used for cylinder fitting. However, the computation will be huge. To alleviate the computation burden, object primitives were further locally optimized—that is, the similar object primitives will be fused together. According to the optimized objected primitives and the separated branches, the topology relationship among each primitive can be built using the graph structure. In the graph construction, the nodes are the centers of each primitive. Furthermore, each object primitive will be fitted as a cylinder. To revise the local abnormal cylinder, a self-adaptive optimization method was proposed. Finally, the constructed tree model was further optimized globally based on prior knowledge. To sum up, six main steps are included in this paper, namely ① wood points segmentation based on the joint neighboring growing; ② single-branch separation based on spatial connectivity analysis; ③ local object primitive self-adaptive constraint adjustment; ④ branch topological relation construction based on graph structure; ⑤ local self-adaptive repair and optimization of branch model; and ⑥ global optimization of tree model guided by prior knowledge.

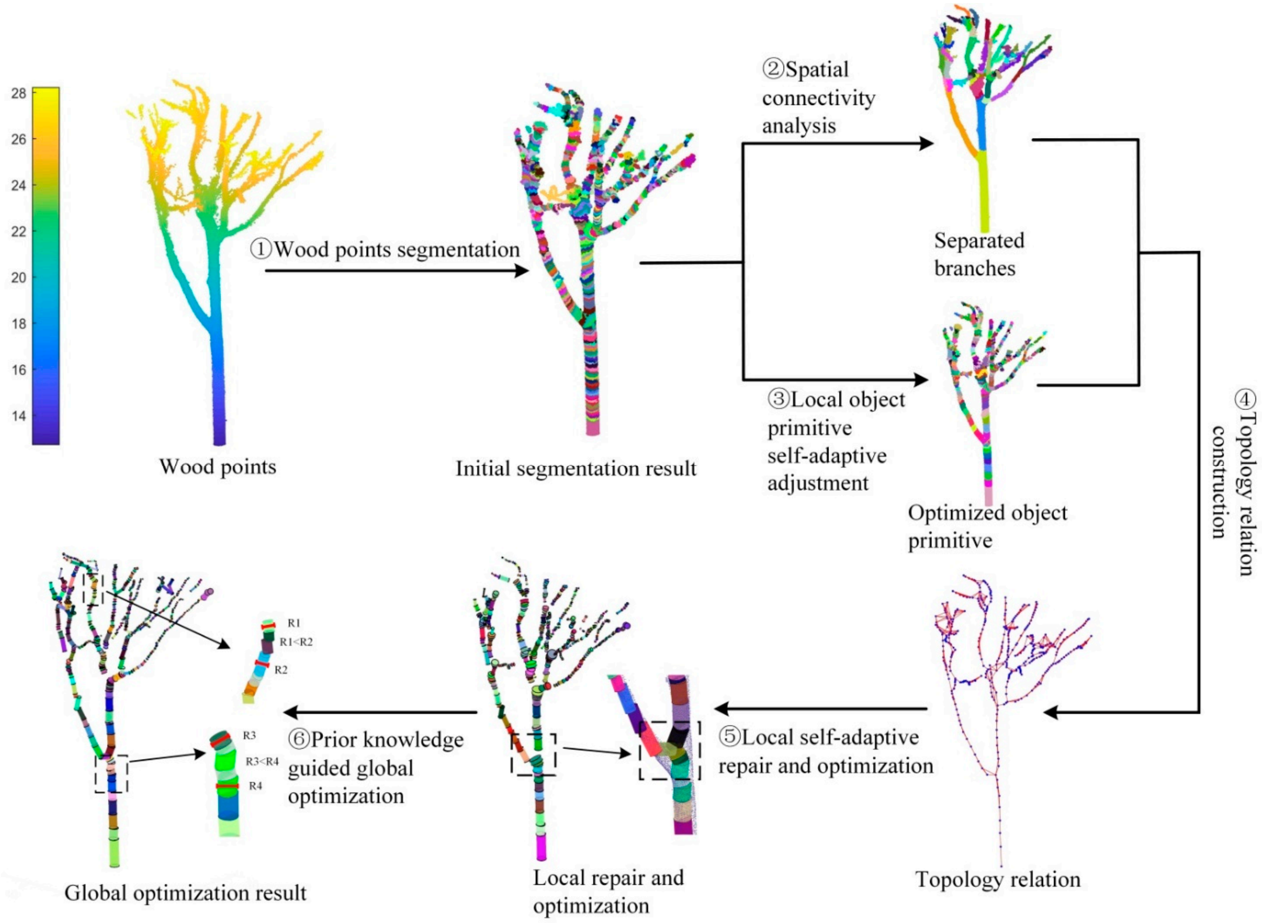

28
26
24
22
20
18
16
14
Wood points
①Wood points segmentation
Initial segmentation result
②Spatial connectivity analysis
Separated branches
③Local object primitive self-adaptive adjustment
Optimized object primitive
④Topology relation construction
Topology relation
⑤Local self-adaptive repair and optimization
Local repair and optimization
⑥Prior knowledge guided global optimization
Global optimization result
R1
R1<R2
R2
R3
R3<R4
R4

Figure 1. Flowchart of the proposed method.

2.1. *Wood Points Segmentation Based on the Joint Neighboring Growing*

To build the individual tree model, the local tree points should be first fitted into cylinders. Then, by combining these cylinders together, the three-dimensional tree structure model can be obtained. Thus, the wood points should be first segmented as objective primitives to conduct further cylinder fitting. In this paper, a joint neighboring growing method was proposed. In this segmentation method, only the growing neighboring points can be seen as an object primitive. The growing starts from the stem base points, which are points lower than 0.7 m from the tree root. In each time of growing, the K nearest neighbors of each point within the starting points are found and merged together as the nearest neighboring points. This procedure can be defined as Equation (1):

$$\left\{set^{k}\right\} = \left\{p \,\middle|\, \bigcup_{i=1\cdots n} neighbors^{k}[p_i], p_i \in \{startingpts\}\right\} \tag{1}$$

where $\left\{set^k\right\}$ is the K nearest neighboring points set. Note that in terms of TLS point clouds, the point density is generally non uniform due to the scan angle of the laser scanner. As shown in Figure 2a, the point density of the left blue points is higher than that of the right blue points. As a result, when detecting the K nearest neighbors of the blue points according to Equation (1), the orange points are found. Obviously, these points do not meet the acquirement that the orange points should be able to fit as a cylinder. To solve this problem, a joint growing strategy was proposed here. On the basis of the detected nearest neighbors, a fixed neighboring radius constraint is applied. That is, only the neighboring points within the r radius are detected as the growing neighboring sets. This is defined as Equation (2):

$$\left\{set_r^k\right\} = \left\{p \middle| \|p, p^c\| \le r, p \in \left\{set^k\right\}\right\} \tag{2}$$

where p^c is the center point of $\{startingpts\}$, $\|p, p^c\|$ means the distance between two points, and r is the radius. In this paper, r is calculated as the mean value of top ten percentile of the points within $\{startingpts\}$ to center p^c. As shown in Figure 2b, red points are excluded successfully under the constraint of the fixed radius. Green points are the final growing neighboring sets. Apparently, these green points will be easy to fit as a cylinder.

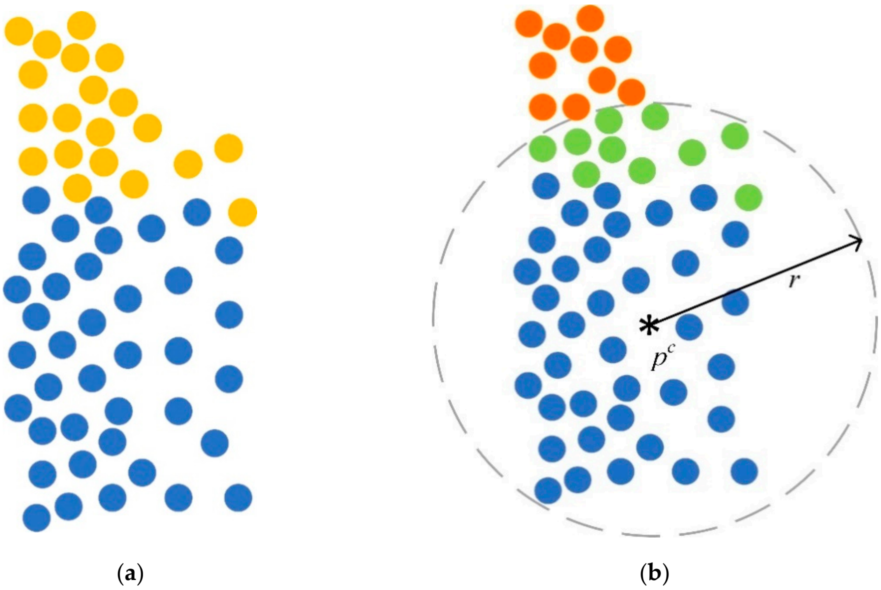


Figure 2. Neighbors growing based on joint K-neighbors and fixed radius. (**a**) Neighbors growing based on K-neighbors; (**b**) neighbors growing under the constraint of the fixed radius. In (**a**), blue points are the starting points, while orange points are the growing points using K nearest neighbors. In (**b**), green points are the final growing points for the blue points under the constraint of a fixed radius. Red points are the excluded points.

2.2. Single-Branch Separation Based on Spatial Connectivity Analysis

To build the correct topology relation among the branches, each branch should be first separated. In this paper, we realized this based on the spatial connectivity of the neighboring points set. As shown in Figure 3, P1 represents the growing starting point set. According to the method mentioned in Section 2.1, in each time of growing, the neighboring point set will be acquired as the $\{Set_i\}$ shown in Figure 3. To separate each branch, spatial connectivity analysis was conducted to the neighboring points set. The points within $\{Set_i\}$ were first voxelized as a series of voxels. By conducting eight connectivity analysis to these voxels, the number of connected components can be obtained. As shown in Figure 4, it is easy to find that the number of connected components in Figure 4a is 1, while the number of connected components in Figure 4b is 2.

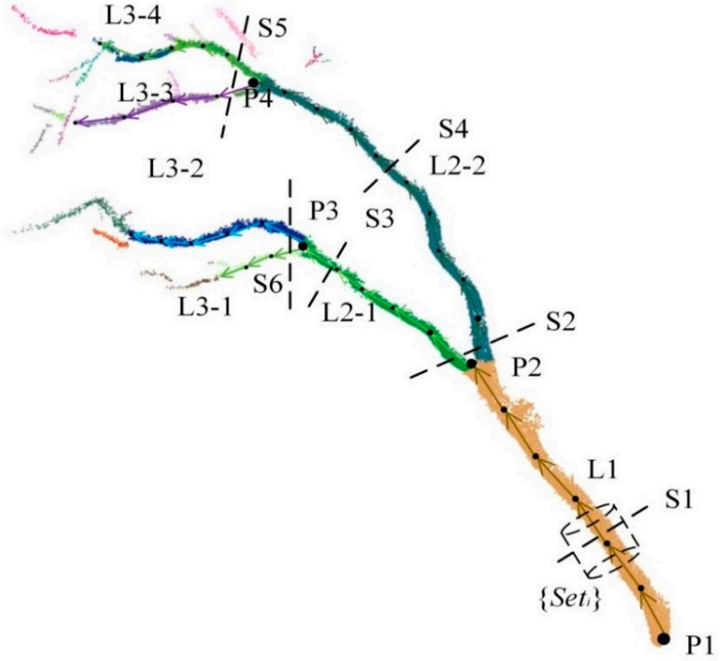


Figure 3. Branches separation based on the spatial connectivity of neighboring points set. $\{Set_i\}$ indicates the neighboring point set. Li represents the separated branches. Si along with the dotted line means the spatial connectivity analysis to the neighboring point set. Pi is the location where connectivity changes. Arrow means the growing direction.

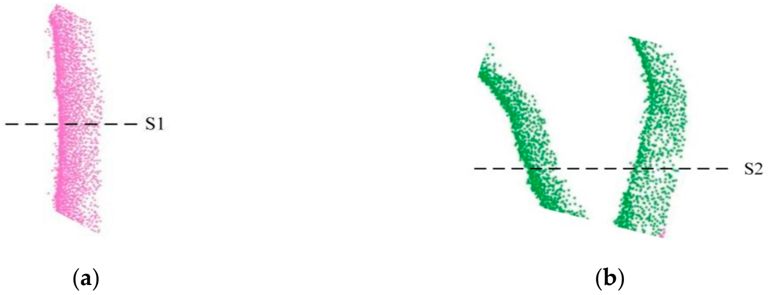


Figure 4. Spatial connectivity analysis to the neighboring point set. (**a**) The number of connected component is 1; (**b**) the number of connected components is 2.

In this paper, the growing starts from the starting points. In each time of growing, the spatial connectivity of neighboring points set was conducted. If the connectivity of the set of adjacent points does not change, the neighboring point sets belong to the same branch, such as L1 shown in Figure 3. When the connectivity of the set of adjacent points changes, such as P2, the branches should be separated. As shown in Figure 4b, it can be found that the number of connected components is 2. From P2, the branches will be labeled as L2. According to the connected components, the branches will be further labeled as L2-1 and L2-2 as shown in Figure 3. Thereafter, the growing will continue along each branch. By conducting the spatial connectivity analysis, the branches were labeled correspondingly.

2.3. Local Object Primitive Self-Adaptive Constraint Adjustment

According to the method described in Section 2.1, the wood points can be segmented as a series of object primitives as shown in Figure 5a. From Figure 5a, it can be found that the neighboring points are clustered as a sequence of individual object primitives. These object primitives can be directly used for cylinder fittings. However, this procedure will involve too much calculation and will be time-consuming since there are too many object primitives. Moreover, it is also not convenient for the lightweight expression of the tree structure model. To alleviate the computation burden, in this paper, we first adjusted the object primitives locally.

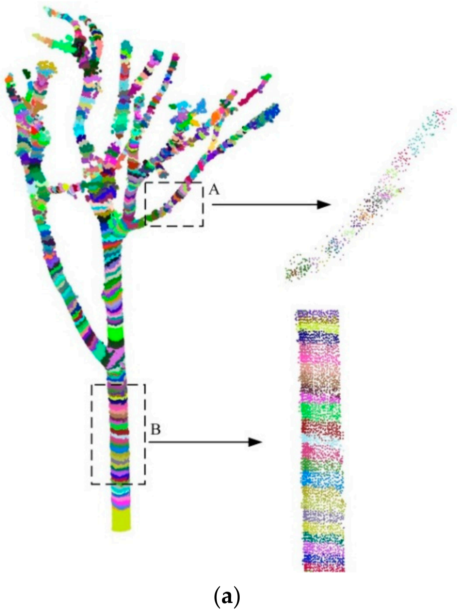


Figure 5. Local object primitive self-adaptive constraint adjustment. (**a**) Segmented object primitives based on the joint neighboring growing; (**b**) result of object primitive self-adaptive adjustment. Different primitives in (**a**,**b**) are rendered in different colors.

As shown in Figure 6, $\{Set_1\}$, $\{Set_2\}$, and $\{Set_3\}$ are three successive points sets. p_1^c, p_2^c, and p_3^c are their corresponding center points. From p_1^c to p_2^c, the direction vector $\overrightarrow{p_1^c p_2^c}$ can be obtained. Similarly, the direction vector $\overrightarrow{p_2^c p_3^c}$ can also be obtained. As shown in Figure 6, θ is the angle between $\overrightarrow{p_1^c p_2^c}$ and $\overrightarrow{p_2^c p_3^c}$. Meanwhile, $p_1^c p_2^c$ can be seen as the centerline of point sets $\{Set_1\} \cup \{Set_2\}$ and $p_2^c p_3^c$ can be seen the centerline of point sets $\{Set_2\} \cup \{Set_3\}$. r_{12} and r_{23} are the fitted radiuses of point sets $\{Set_1\} \cup \{Set_2\}$ and $\{Set_2\} \cup \{Set_3\}$, which can be calculated as the average distance between each point within $\{Set_1\} \cup \{Set_2\}$ and $\{Set_2\} \cup \{Set_3\}$ to its corresponding centerlines. In this paper, angle θ and radius change rate δ were combined to determine whether the point sets can be merged together. θ and δ are defined as Equation (3):

$$\begin{cases} \theta = < \overrightarrow{p_1^c p_2^c}, \overrightarrow{p_2^c p_3^c} > \\ \delta = abs(r_{23} - r_{12})/r_{12} \end{cases} \tag{3}$$

In this paper, if $\theta < 15^\circ \,\&\, \delta < 0.15$, point sets $\{Set_1\}$, $\{Set_2\}$, and $\{Set_3\}$ can be merged together. It means only the point sets with similar radius and directions can be fused together. For better cylinder fitting, the max number of point sets fusing is set to 5. That is, at most five successive point sets can be fused together. In so doing, the point sets fitting error as a cylinder will not be larger.

Figure 5 shows the comparison of object primitive before and after self-adaptive constraint adjustment. Figure 5a is the segmented object primitives based on the joint neighboring growing. Figure 5b is the result of object primitive self-adaptive adjustment. From the comparisons between 'A' and 'C' and 'B' and 'D' shown in Figure 5a,b, it can be found that the similar neighboring point sets are fused effectively. The object primitives after fusing show a better shape of cylinder, which will lead to a better cylinder fitting result.

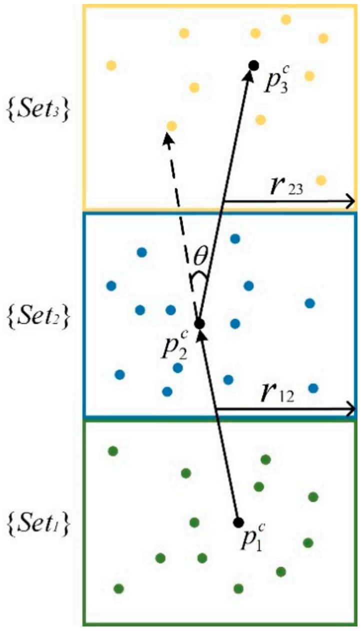


Figure 6. Sketch map of neighboring point sets fusing.

2.4. Branch Topological Relation Construction Based on Graph Structure

To construct the correct tree structure model, the topological relation among different branches should be built. In this paper, we adopted the graph structure to build the branch topological relation. In the graph, two sets are concluded, namely node and edge, which is defined as Equation (4):

$$G = (V, E) \tag{4}$$

where V represents the node set, while E represents the edge set. In this paper, v_i is the center point of each object primitive after locally adjustment. $e_{i,j}$ is the edge between two nodes. In this paper, $e_{i,j}$ is defined as Equation (5):

$$e_{i,j} = \begin{cases} 1\ , & if \quad \begin{array}{l} v_j \in \{neighbors(v_i)\} \\ v_i \in \left\{branch^l\right\} \& v_j \in \left\{branch^l\right\}, i \neq j \end{array} \\ 1\ , & elseif \quad \begin{array}{l} v_j \in \{neighbors(v_i)\} \\ v_i \in \left\{endpoint^l\right\} \& v_j \in \{branch^m\}, l \neq m \end{array} \\ 0\ , & otherwise \end{cases} \tag{5}$$

As defined in Equation (5), if $e_{i,j}$ is existed, two requirements should be met. One is that v_j is one neighboring point of v_i. Meanwhile, v_i and v_j belong to the same branch. The other one is v_i is one endpoint of the lth branch. Meanwhile, v_j is one point of another branch. In so doing, the built graph structure will be more like a tree skeleton. In Section 2.2, this paper has separated each branch. Thus, it will be easy to determine whether v_i and v_j belong to the same branch or not. Note that to limit the number of edges, only six nearest neighbors were selected for each node in this paper. Moreover, if the length of $e_{i,j}$ is long, $e_{i,j}$ will also be eliminated. Figure 7 shows the graph structure for the tree. From Figure 7, it can be found that the graph can reflect the topology relation among the object primitives. According to the graph structure, the following tree model can be optimized self-adaptively, which will be described in Sections 2.5 and 2.6.

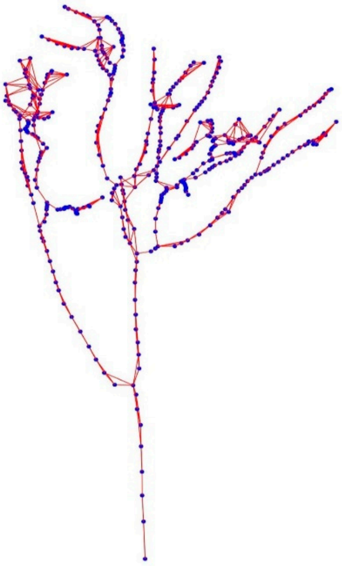
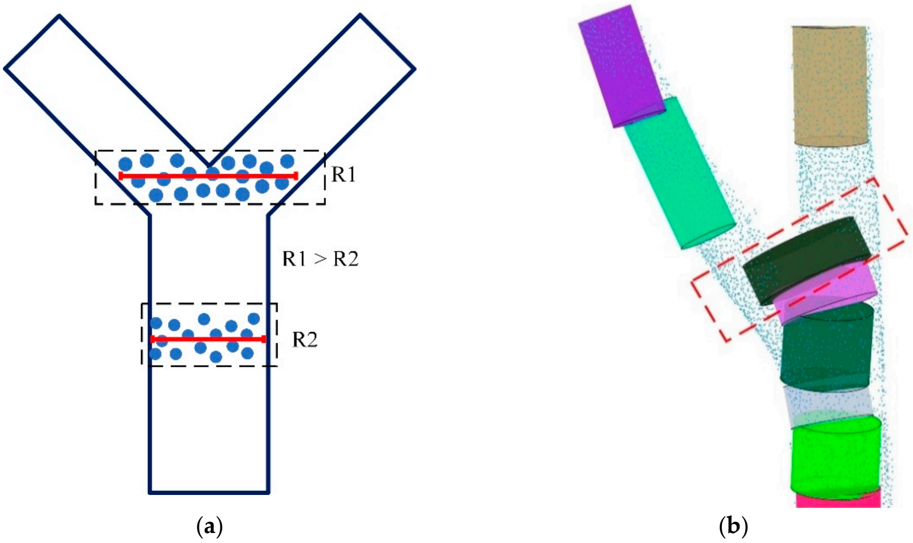


Figure 7. Graph structure for the tree.

2.5. Local Self-Adaptive Repair and Optimization of Branch Model

After object primitives adjusted, each object primitive can be fitted as a cylinder by finding the central axis and calculating the diameter. The radius can be calculated as the mean value of distance between each point within the object primitive to the central axis. However, the cylinder fitting errors always occur when encountering the branch cross as shown in Figure 8. In this paper, the neighboring points will be clustered as object primitives. Thus, the diameter of the fitted cylinder for each object primitive can be calculated as two times of the mean value of distance between each point to the central axis. However, when encountering with the cross of branches the calculated diameter always exists error. As shown in Figure 8a, R1 is obvious larger than R2. As a result, the fitted cylinder based on the larger diameter will be abnormal as shown in Figure 8b. To obtain better tree modeling result, the abnormal diameters for fitting cylinders should be revised.



(**a**) (**b**)

Figure 8. Abnormal fitting for local object primitive. (**a**) Local abnormal fitted diameter; (**b**) abnormal fitted cylinder. In (**a**), it can be found that R1 is obvious larger than R2. The red frame represents the abnormal cylinder.

2.5.1. Abnormal Fitted Cylinder Optimization

To obtain a better tree modeling result, the abnormal fitted cylinder should be optimized. This paper optimized the tree model mainly based on the priori knowledge of tree growth. Figure 9a shows the diameter changes along one same branch. Obviously, R1 > R2 > R3. Figure 9b shows the diameter changes among different branches. Likewise, R1 > R2 > R3. Thus, it can be concluded that under natural conditions, branches closer to the root tend to have larger diameters.

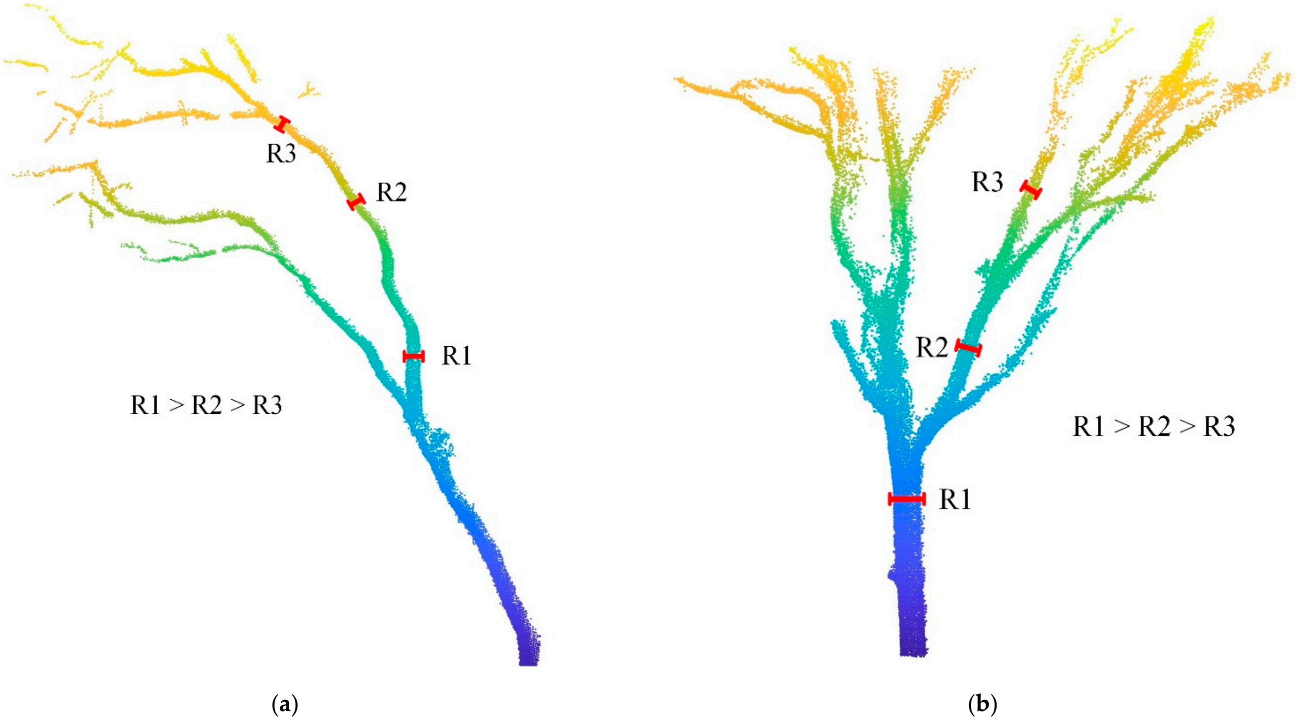


Figure 9. Variation in diameter of branches under natural conditions. (**a**) Variation in diameter of one branch; (**b**) variation in diameter of different branches.

According to the separated branches mentioned in Section 2.2 and topology relation among branches built in Section 2.4, the abnormal cylinder can be detected as follows. Starting from the root of stem, every branch was traversed along the growing direction. If the diameter (R) of fitted cylinder is larger than that of last neighboring fitted cylinder (R_{last}), the current cylinder will be detected as an abnormal cylinder and its diameter (R) should be revised as Equation (6):

$$R = \begin{cases} mean(Rlast, Rnext)\ if\ Rnext < Rlast \\ \eta * Rlast\ if\ Rnext \geq Rlast \end{cases} \tag{6}$$

In this paper, R_{next} represent the diameter of next neighboring cylinder. η is the adjustment coefficient. In this paper, η is set to 0.99. It means the diameter of the revised cylinder should be close but smaller than its nearest cylinder.

Figure 10 shows tree model comparison before and after local optimization. From Figure 10, it can be found that after local optimization, the abnormal cylinders were revised successfully. For example, R1 > R2 in Figure 10a was revised as R1 < R2 in Figure 10b; R3 > R4 in Figure 10a was revised as R3 < R4 in Figure 10b.

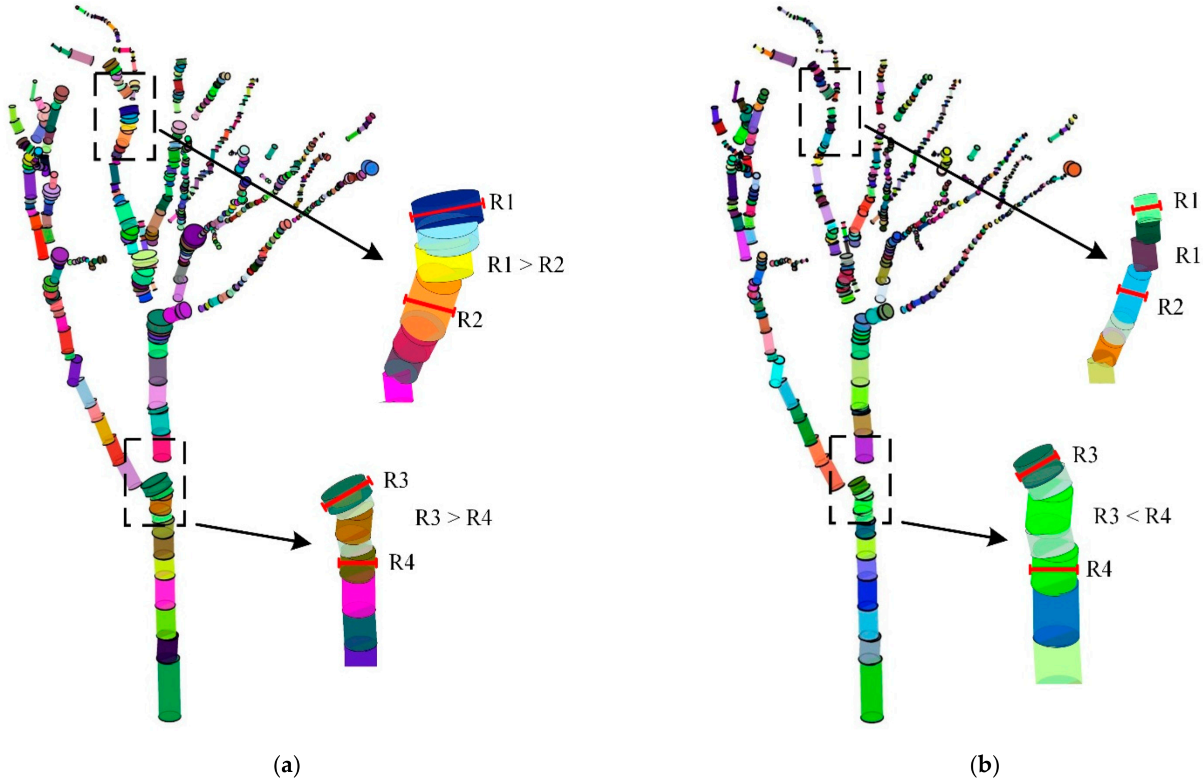
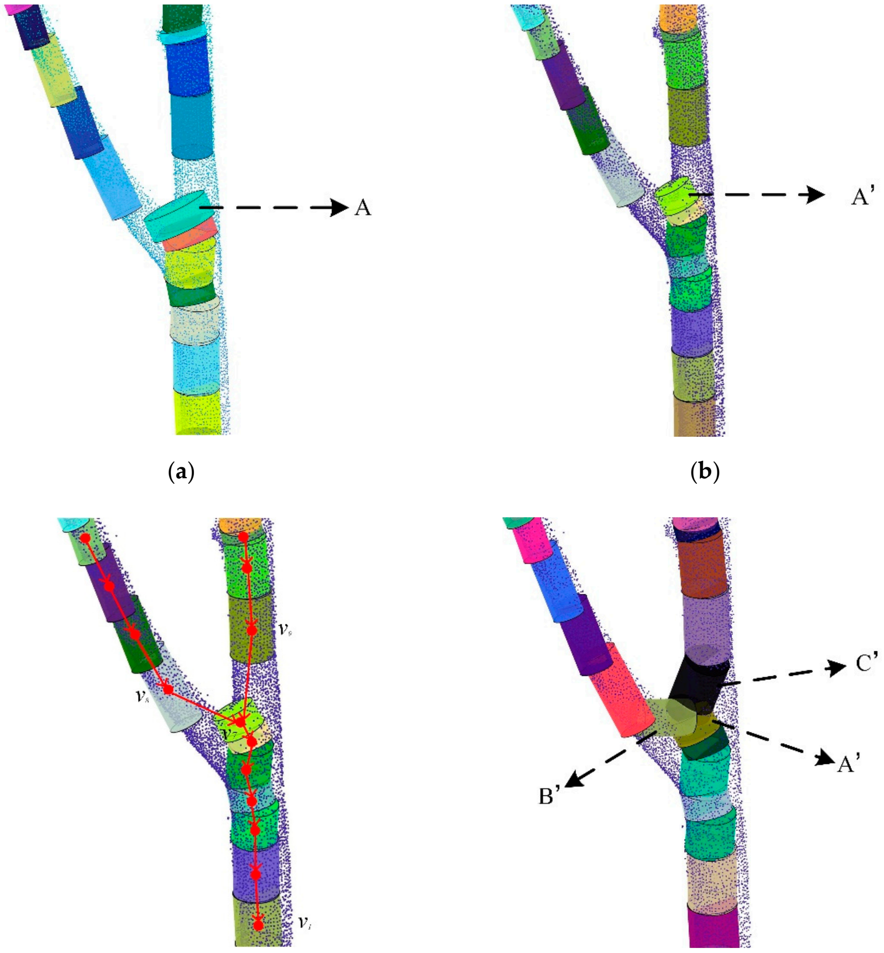


Figure 10. Branch model comparison before and after local optimization. (**a**) Branch model before optimization; (**b**) branch model after optimization. R1, R2, R3, and R4 are the diameters of the fitted cylinders.

2.5.2. Self-Revision of Branch Crosses

As shown in Figure 11a, 'A' is the abnormal cylinder that needs to be optimized. After local optimization as mentioned above, the abnormal cylinder can be adjusted as shown in Figure 11b. From Figure 11b, it is easy to find that the diameter of the abnormal cylinder was revised effectively. However, the geometry of the branch cross is destructed. In other words, the constructed tree model is not connected at the branch cross. To solve this problem, this paper mainly used graph path analysis to repair the gap caused by the abnormal cylinder.

As shown in Figure 11c, each cylinder can be seen as an object primitive. It is because the cylinders are built from object primitives. The center points that can be seen as nodes in graph within each cylinder can be obtained. Thereafter, according to Equations (4) and (5), the graph structure for the tree can be constructed. To repair the branch cross, the neighboring cylinders of the revised cylinder 'A" as shown in Figure 11b should be found. In this paper, it was realized by using the shortest path analysis. According to the constructed graph, each node will have a shortest path to the tree root based on the Dijkstra algorithm. For example, in Figure 11c, the shortest path from v_7 to v_1 is $SP\{v_7 \rightarrow v_1\}$. The shortest path from v_8 to v_1 is $SP\{v_8, v_7 \rightarrow v_1\}$. The shortest path from v_9 to v_1 is $SP\{v_9, v_7 \rightarrow v_1\}$. Thus, it is easy to find that v_8 and v_9 are the neighboring nodes of v_7. To repair the gap caused by the abnormal cylinder, two cylinders between v_8 and v_7, v_9 and v_7, should be added. The diameter of added cylinder between v_8 and v_7 can be calculated the mean value of diameters of the cylinders corresponding to v_8 and v_7. Similarly, the diameter of added cylinder between v_9 and v_7 can also be calculated. Figure 11d shows the self-revision result after cylinder repair. It can be found that the cylinders 'B" and 'C" filled the gaps, which are formed by abnormal cylinder successfully.



(**a**) (**b**)

(**c**) (**d**)

Figure 11. Self-revision of branch crosses. (**a**) Local abnormal cylinder; (**b**) cylinder after optimization; (**c**) shortest path analysis based on the constructed graph structure; (**d**) branch crosses after self-revision. In (**a**), ‘A’ represent the abnormal cylinder. In (**b**), ‘A″ represents the optimized cylinder. In (**d**), ‘B″ and ‘C″ represent the cylinders filled the gap.

2.6. Global Optimization of Tree Model Guided by Prior Knowledge

Figure 10b shows the local abnormal cylinder optimization result. From Figure 10b, it can be found that after local optimization, the abnormal cylinders can be detected and revised effectively. However, at this time, the overall shape of the model still does not conform to the trend of the branch model gradually thinning along the growth direction of the tree. This is because in Figure 10b, only local optimization is carried out without considering the overall shape change trend. Therefore, further overall optimization of the optimization results in Figure 10b is required in this paper. Specific overall optimization method is described as follows. According to the separated branches and topology relation among them, the branches can be set as different levels. The branch closer to the tree root is set as lower level. As shown in Figure 3, the branch closet to the root is set as L1 level. Along the growth direction, the level of branch increased gradually. To optimize the branch model globally, the cylinder within each branch was traversed. In general, for the diameter of different level of branches, the branch closer to the root tends to have larger diameter. Thus, if the diameter of one cylinder that forms the branch of the next level is larger than the average diameter of all cylinders forming the last level, the corresponding cylinder model will be detected as an abnormal cylinder and the diameter of the cylinder will be adjusted. This procedure can be defined as Equation (7):

$$if \quad R_L(k) > \sum_{i=1}^{n} R_{L-1}(i)/n \qquad R_L(k) = \max\left(R^n_{L-1}, R^{n-1}_{L-1}, R^{n-2}_{L-1}\right) \tag{7}$$

where $R_L(k)$ represents the diameter of the kth cylinder within the Lth level of branch. $R_{L-1}(i)$ represents the diameter of the ith cylinder within the L-1th level of branch. n is the number of cylinder forming the L-1th level of branch.

3. Result

To evaluate the performance of the proposed method, 29 field datasets obtained from three forest sites, namely Peru, Indonesia, and Guyana were tested (Tanago et al. [37]). The datasets were acquired using the RIEGL VZ-400 terrestrial laser scanner, of which the scan resolution is 0.06° and the scan range is 360° in azimuth and 100° in zenith. The characteristics of the 29 datasets were tabulated in Table 1. From Table 1, it can be found that these 29 datasets are from three different type of forest. The mean tree elevation is from 22 m to 312 m. The mean stem density is also different from each other. The minimum mean stem density is 516 stems/ha, while the maximum mean stem density is 1314 stems/ha. Moreover, the mean DBH of different forest type is also different. Figure 12 shows 12 tree samples selected from the three forest sites. Obviously, they own different morphological characteristics. Thus, these 29 datasets are representative for validating the effectiveness of the proposed method.

Table 1. Characteristics of the 29 datasets.

	Peruvian Site	**Indonesian Site**	**Guyanese Site**
Number of plots	9	10	10
Forest type	Lowland tropical moist Terra firme forest	Peat swamp forest	Lowland tropical moist forest
Region	Madre de Dios. South western Amazon	Mentaya River (Central Kalimantan)	Vaitarna Holding's concession

For the 29 datasets, the geometry of the stem, buttresses, and branches of each harvested tree was measured. Thus, the reference volume of each harvested tree can be calculated by summing the volumes of stem, buttresses, and branches. Since the tree model is built by connecting a series of cylinders, the volume of the built tree model can be calculated as the sum of these cylinders. Obviously, the tree modeling accuracy can be evaluated by comparing the volumes of referenced harvest tree and the built tree model. Moreover, this paper also estimated the tree heights and DBHs for each individual trees (Table 2).

Table 2. Experimental results of the proposed method for 29 samples.

Sample	**Species**	**Volume (m^3)**	**Height (m)**	**DBH (cm)**
PER01	*Buchenavia macrophylla*	41.934	38.957	137.6
PER02	*Dacryodes peruviana*	10.385	26.688	76.8
PER03	*Couratari macrocarpa*	7.799	31.878	77.4
PER04	*Couratari macrocarpa*	5.956	34.624	66.2
PER05	*Sloanea eichleri*	25.91	35.053	108
PER06	*Pterygota amazonica*	21.353	41.837	115.4
PER07	*Pterygota amazonica*	14.111	43.997	117
PER08	*Pseudopiptadenia suaveolens*	20.144	43.231	91.4
PER09	*Nectandra longifolia*	7.82	34.012	67.1

Table 2. *Cont.*

Sample	Species	Volume (m^3)	Height (m)	DBH (cm)
IND01	*Tetramerista glabra*	1.578	23.251	41.5
IND02	*Tetramerista glabra*	2.918	25.214	59.8
IND03	*Tetramerista glabra*	4.545	23.758	66.8
IND04	*Parastemon urophyllus*	1.751	26.288	38.3
IND05	*Cratoxylon glaucum*	0.974	21.446	34.6
IND07	*Shorea*	15.859	36.651	89.6
IND08	*Aglaia rubiginosa*	3.732	26.389	61.3
IND09	*Diospyros evena*	4.717	23.373	51
IND10	*Shorea teysmanniana*	2.697	24.999	49.1
IND11	*Shorea*	12.869	36.457	79.8
GUY01	*grandiflora*	13.207	32.261	88.3
GUY02	*jupunba*	5.646	31.781	63.9
GUY03	*grandiflora*	6.078	29.138	60.3
GUY04	*grandiflora*	6.527	28.476	62.6
GUY05	*grandiflora*	5.98	30.017	66.4
GUY06	*grandiflora*	6.382	31.484	70.5
GUY07	*grandiflora*	12.455	33.996	95.8
GUY08	*grandiflora*	8.661	28.924	75.9
GUY09	*coutinhoi*	16.817	35.051	95.2
GUY10	*falcata*	8.506	27.893	65.4

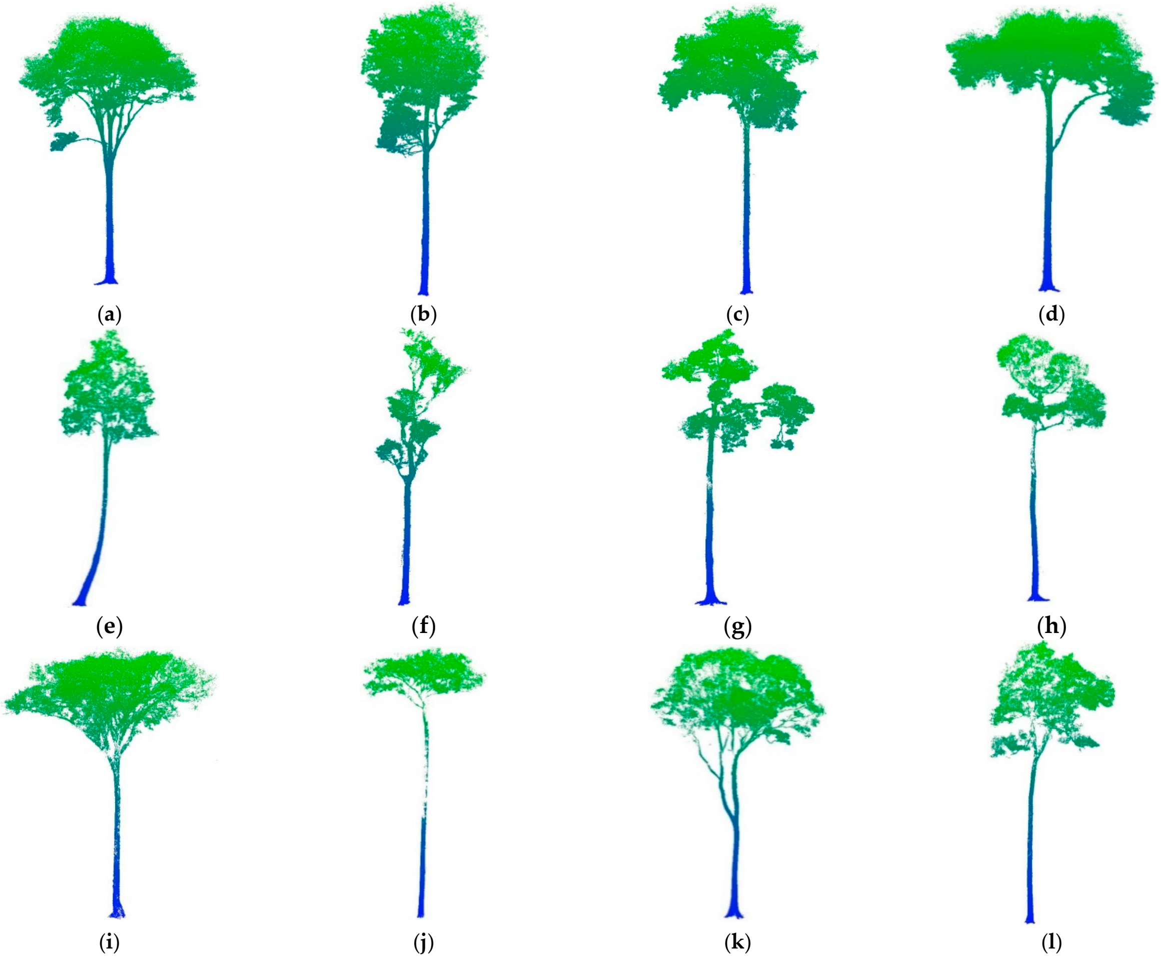


Figure 12. Tree samples for the three forest sites. (**a**–**d**) tree samples from Peruvian site; (**e**–**h**) tree samples from Indonesian site; (**i**–**l**) tree samples from Guyanese site.

Figures 13–15 shows the deviations of volume, height, and DBH for each sample according to the referenced values. In our paper, volume is calculated as the sum of a series of cylinders forming the tree. Tree height is calculated based on the highest and lowest points within the tree points. DBH is calculated using the method proposed by Wang et al. [38]. In this DBH estimation method, Fourier series and circle fitting are combined. As a result, the root mean square error of DBH can be improved by 12.4%. From Figure 13, it can be found that there are 21 out of 29 QSM volume deviations that are larger than -2 m^3 and smaller than 2 m^3. It means that 72.4% testing samples can achieve accurate volume estimation results. In other words, the QSMs constructed by the proposed method are also similar to the referenced ones. Thus, it can be concluded that the proposed method can achieve good modeling effect. Figure 14 shows the tree height deviation corresponding to the referenced values. From Figure 14, it can be found that the tree heights measured by terrestrial laser scanner tend to be lower. It is because the treetops cannot be accurately detected by terrestrial laser scanner due to the limit of the vertical scan angle. As a result, most of the tree height deviations are negative. Figure 15 shows the DBH deviation for each sample. From Figure 15, it can be found that the DBH estimation results are similar to the referenced values. Most of the DBH deviations are smaller than 10 cm. Thus, it can be concluded that terrestrial laser scanner can achieve good DBH measuring results.

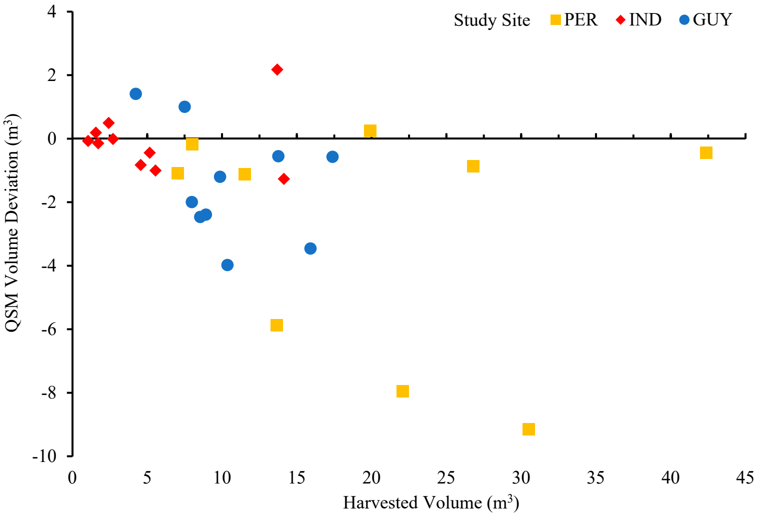


Figure 13. Volume deviation for each sample.

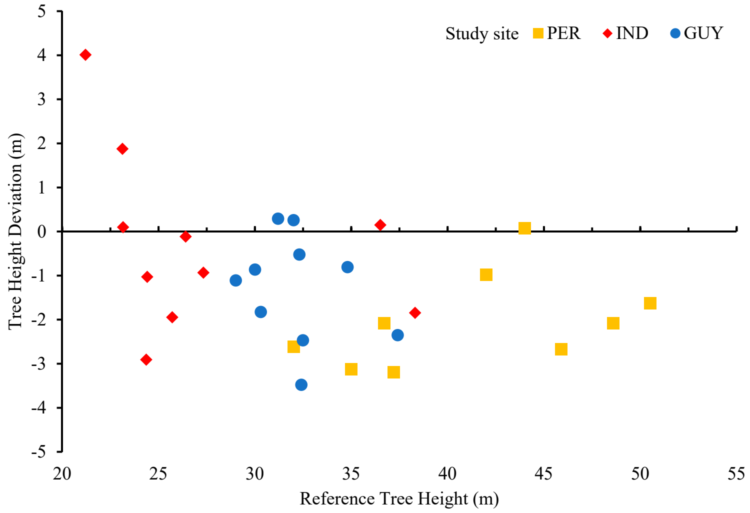


Figure 14. Tree height deviation for each sample.

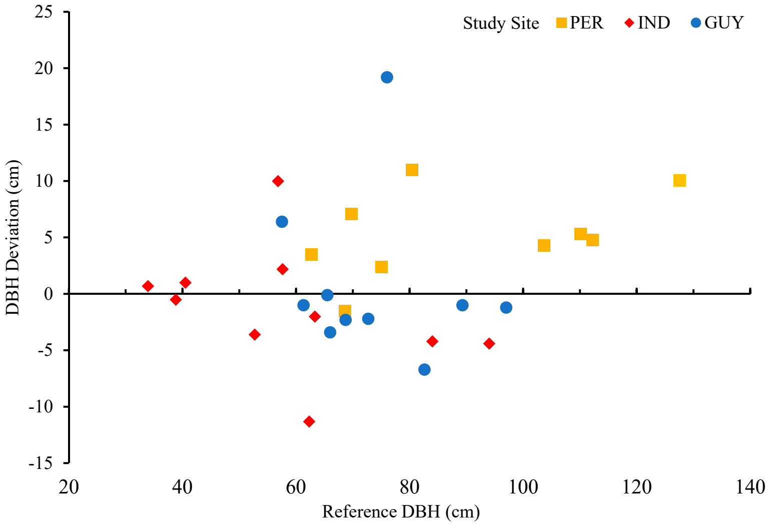


Figure 15. DBH deviation for each sample.

Comparing with tree height or DBH, tree volume can better reflect the performance of the tree modeling method. It is because the tree volume can be calculated as the sum of a series of cylinders forming the tree. If the tree modeling effect is not good, the volume of the constructed tree model cannot be similar to the harvested volume. Thus, in the following of our paper, tree volume was mainly used for accessing the performance of our proposed method. Five accuracy indicators, namely mean deviation (*md*), relative mean deviation (*rmd*), root mean square error (*rmse*), relative root mean square error (*rmse*), and concordance correlation coefficient (*ccc*), were adopted for quantitative analysis, which are defined as Equations (8)–(12).

$$md = \sum_{i=1}^{n} (Vol_i - Vol_i^r)/n \tag{8}$$

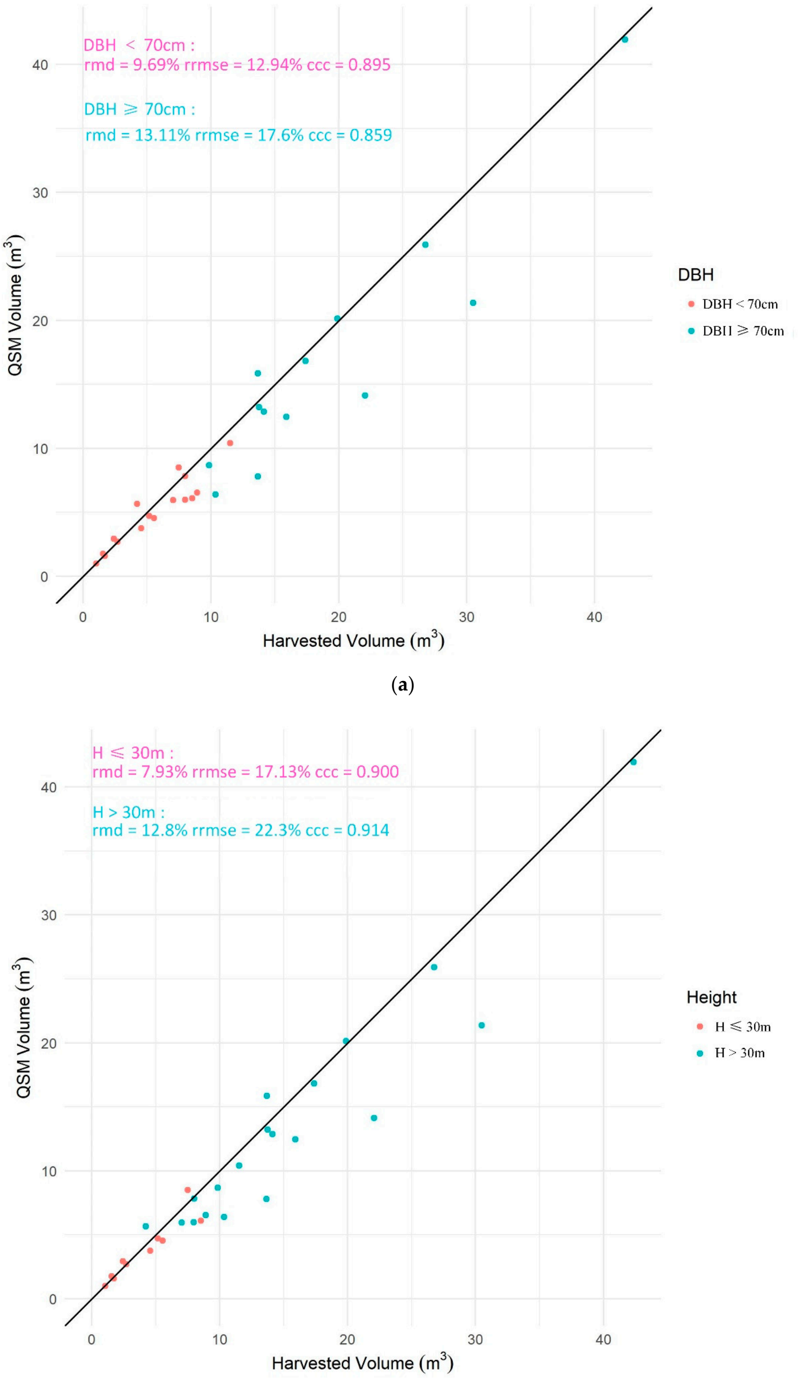
$$rmd = md/\overline{Vol^r} \tag{9}$$

$$rmse = \sqrt{\sum_{i=1}^{n} (Vol_i - Vol_i^r)^2/n} \times 100\% \tag{10}$$

$$rrmse = rmse/\overline{Vol^r} \times 100\% \tag{11}$$

$$ccc = \frac{2 * \sum_{i=1}^{n} (Voli - \overline{Vol})(Vol_i^r - \overline{Vol^r})/n}{\frac{1}{n}\sum_{i=1}^{n} (Voli - \overline{Vol})^2 + \frac{1}{n}\sum_{i=1}^{n} (Vol_i^r - \overline{Vol^r})^2 + (\overline{Vol} - \overline{Vol^r})^2} \tag{12}$$

where Vol_i is the volume of the ith sample, Vol_i^r is its corresponding harvested volume, $\overline{Vol^r}$ is the average of harvested volume, and $\overline{Vol}$ is the average of estimated volume. From Figure 13, it can also be found that the largest QSM volume deviations belong to the study site of PER. Combining with the results shown in Figures 14 and 15, it is easy to find than the trees in PER tend to be larger since these trees tend to have higher tree heights and larger DBHs. To further analyze this, this paper calculated *rmd*, *rrmse*, and *ccc* of trees with different DBH and tree height. Figure 16a shows accuracy metrics of trees with different DBHs. From Figure 16a, it can be found that *rmd* and *rrmse* of the trees with DBH smaller than 70 cm are obviously smaller than the ones of the trees with larger DBHs. A similar result can be found in Figure 16b. *rmd* and *rrmse* of the trees with height lower than 30 m are obviously smaller than the ones of the trees with higher heights. Thus, it can be concluded that the proposed method tends to achieve better modeling results for the trees with smaller DBH and lower tree height. In addition, no matter which kind of trees, the proposed method can achieve good concordance correlation coefficient (*ccc*). All of the concordance correlation coefficients are larger than 0.85. In statistics, *ccc* indicates the agreement between two variables. In Figure 16, the two variables are QSM volume and harvested volume. Larger *ccc* represents high agreement between the calculated values and referenced values. Thus, it can conclude that the proposed method can achieve good modeling results as a whole.



(**a**)

(**b**)

Figure 16. QSM volume comparison among trees with different DBH and tree height. (**a**) QSM volume comparison among trees with DBH smaller than 70 cm and larger than 70 cm. (**b**) QSM volume comparison among trees with height lower than 30 m and higher than 30 m.

4. Discussion

To access the performance of the proposed method objectively, two classical individual tree modeling methods, namely TreeQSM and AdQSM were adopted for experimental analysis. TreeQSM was developed by Raumonen et al. [23]. In their method, tree points are first segmented as a sequence of cover sets. Thereafter, the neighboring relation among these cover sets can be determined to obtain tree components. Last, tree components are further segmented for fitting cylinders with different radius and length. TreeQSM is implemented in MATLAB and is freely available at https://github.com/InverseTampere/TreeQSM (accessed on 10 March 2022).

AdTree was first developed by Du et al. [12] and was quickly further optimized by Fan et al. [30]. In AdTree, the initial tree skeleton was first built based on the Dijkstra shortest path method. Subsequently, the skeleton was optimized by removing adjacent vertices and edges. Finally, the branch model can be built by fitting a series of cylinders. Fan et al. [30] developed an open tool with an end-user interface for AdTree and named it AdQSM, which can be acquired freely (https://github.com/GuangpengFan/AdQSM, accessed on 10 March 2022).

Table 3 tabulated the volume accuracy metrics of the three methods. The five accuracy indicators as defined as Equations (8)–(12) were calculated for TreeQSM, AdQSM, and ProposedQSM, respectively. From Table 3, it can be found that no matter which accuracy indicator is selected, the proposed method performs the best. In terms of *md*, the mean volume deviation of the proposed method is 1.427 m^3, which is much smaller than that of TreeQSM and AdQSM. Moreover, *rmse* of TreeQSM and AdQSM are also more than two times of that of the proposed method. Similar results can be obtained by comparing *rmd* and *rrmse* with the ones of the proposed method. In terms of *ccc*, the proposed method also performed much better than the other two methods.

Table 3. Volume accuracy metrics of different method. The bold font means the lowest or highest value comparing with other methods.

	TreeQSM	**AdQSM**	**ProposedQSM**
md (m^3)	4.257	2.364	**1.427**
rmse (m^3)	6.732	5.766	**2.887**
rmd	36.45%	20.24%	**12.22%**
rrmse	57.60%	49.40%	**24.70%**
ccc	0.679	0.788	**0.949**

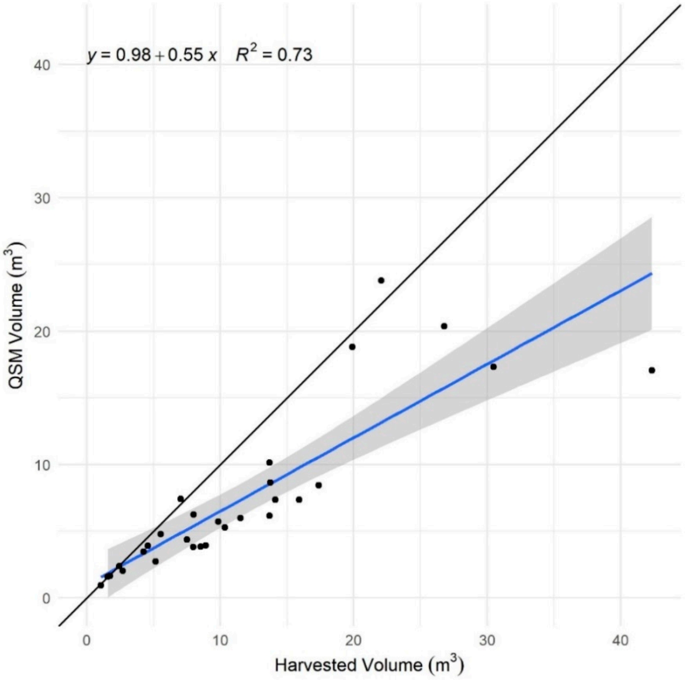
Figure 17 shows the regression analysis for the three methods between the built QSM volume and the harvested volume. For each method, the linear regression model was fitted. Meanwhile, the coefficient of determination (R^2) for each fitted regression model was calculated as Equations (13)–(15).

$$S_{reg} = \sum_{i=1}^{n} \left(f(Vol_i) - \overline{Vol} \right)^2 \tag{13}$$

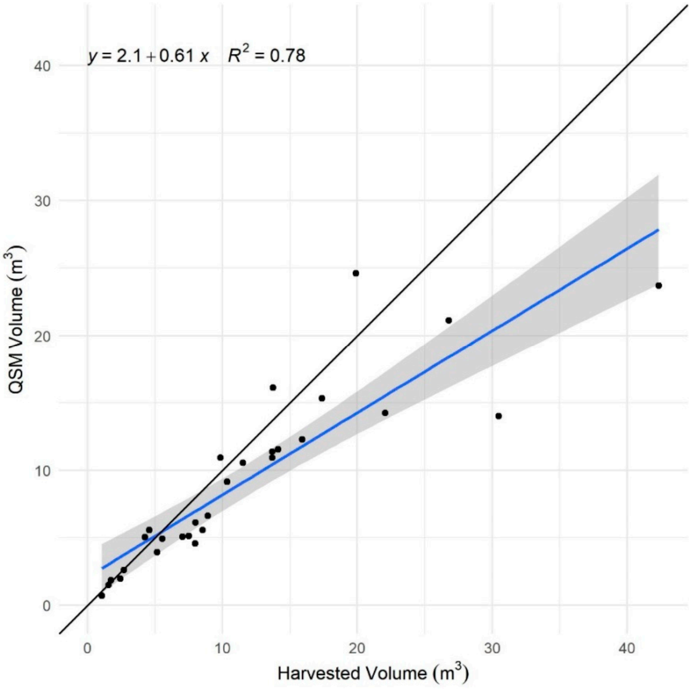
$$S_{tot} = \sum_{i=1}^{n} \left(Vol_i - \overline{Vol} \right)^2 \tag{14}$$

$$R^2 = S_{reg}/S_{tot} \tag{15}$$

where $f(Voli)$ represents the fitted linear regression model. In general, the larger R^2 is, the better fitting can be achieved between the built QSM volume and the harvested volume. In other words, if R^2 is closer to 1, it means a better linear regression model can be achieved between the built QSM volume and the harvested volume. From Figure 17, it is easy to find that R^2 of the proposed method is much higher than that of TreeQSM and AdQSM. Table 4 tabulates the comparison of R^2 toward these three methods. From Table 4, it can be found that the closer S_{reg} to S_{tot} is, the larger R^2 is. This character can also be found from Equations (13)–(15). The closer S_{reg} and S_{tot} means the fitted volume values are close to the real values. Thus, the larger R^2 indicates that a better regression model has been fitted between the QSM volume and the harvested volume.

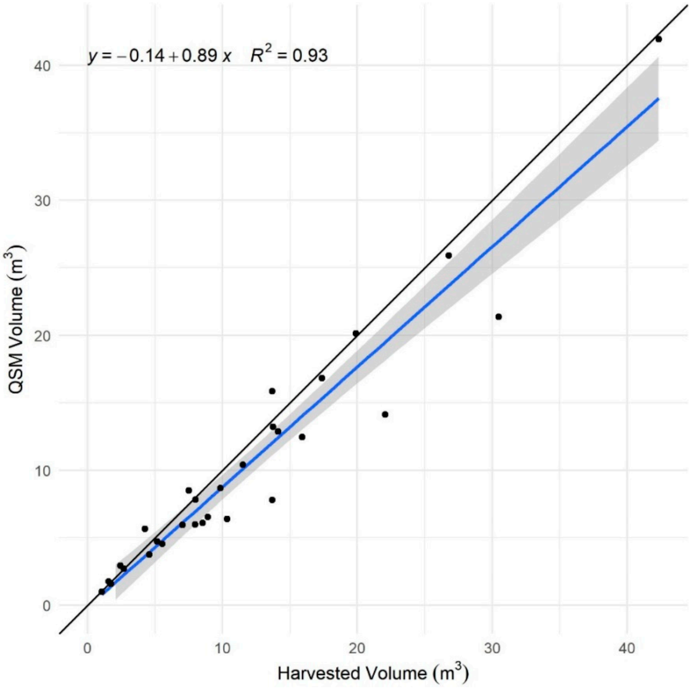


(**a**)



(**b**)

Figure 17. *Cont.*



(**c**)

Figure 17. Regression analysis between the built QSM volume and harvested volume. (**a**) Regression analysis for TreeQSM; (**b**) regression analysis for AdQSM; (**c**) regression analysis for ProposedQSM.

Table 4. Comparison of R^2 toward the three methods.

	Fitted Linear Model	S_{reg}	S_{tot}	R^2
TreeQSM	y = 0.98 + 0.55x	763.57	1048.33	0.73
AdQSM	y = 2.1 + 0.61x	934.93	1195.21	0.78
ProposedQSM	y = −0.14 + 0.89x	1999.39	2151.76	0.93

From Figure 17, it can also be found that the built QSM volumes of the proposed method for the 29 samples are much closer to their corresponding harvested volumes. To further analyze this, in this paper, we compared the volume deviations of the 29 samples for the three methods. Figure 18 shows the box plot of the volume deviations for TreeQSM, AdQSM, and ProposedQSM. From Figure 18, it can be found that range of volume deviation of TreeQSM for the 29 samples is much larger than that of AdQSM and the ProposedQSM. The mean volume deviation of the proposed method is much closer to 0. It means the QSMs built by the proposed method are closer to the referenced tree volumes. Thus, it can be concluded that the proposed method can achieve the best individual tree modeling result.

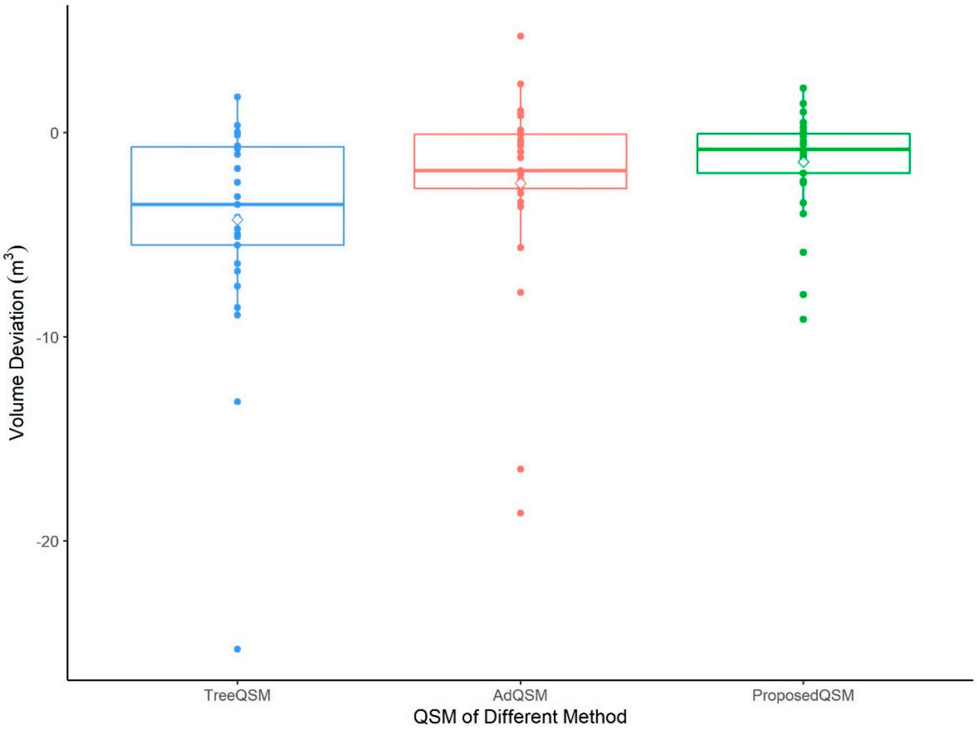


Figure 18. Volume deviations of the 29 samples for the three methods.

5. Conclusions

Individual tree modeling is an important step for forestry investigation and applications. For example, above-ground biomass estimation is generally based on quantitative structural model (QSM). To improve tree modeling accuracy and robustness towards complex forest environments, in this paper, we proposed a self-adaptive three-dimensional tree structure modeling method. In this paper, tree points are first segmented into object primitives using the proposed joint neighboring growing method. Then, the object primitives are further self-adjusted to alleviate computation burden. Subsequently, the object primitives are fitted as a series of cylinders with varying radius and length. Moreover, the built tree model is further optimized locally and globally based on prior knowledge. Twenty-nine field datasets obtained from three forest sites were used for testing the performance of the proposed method. There are 21 out of 29 QSM volume deviations that are smaller than 2 m^3. That is, 72.4% testing samples can achieve accurate volume estimation results. Two other famous tree modeling methods, namely TreeQSM and AdQSM were used for comparing the performance of the proposed method. From comparison, it can be found that no matter which accuracy indicator is selected, the proposed method performs the best. In terms of *md*, the mean volume deviation of the proposed method is 1.427 m^3, which is much smaller than that of TreeQSM and AdQSM. Moreover, *rmse* of TreeQSM and AdQSM are also more than two times of that of the proposed method. In summary, the proposed method can achieve satisfying individual tree modeling result. When encountering trees with different species and tree sizes, the modeling accuracy can be still good. The main reason for this is that the proposed modeling method can be self-adjusted and optimized. Note that the proposed method can only be used for modeling individual trees. When encountering with forest plot with many mixed trees. The individual tree detection methods should be first applied to obtain individual trees. And then, the proposed method can be utilized for modeling. In our future research, we will focus on the tree modeling method toward individual trees with huge data gaps and try to further improve the modeling accuracy. Moreover, we will also conduct research on leaves modeling to realize virtual reality for trees.

Author Contributions: Z.H. and Z.C. conceived the original idea of the study and drafted the manuscript. Z.H. and B.L. contributed to the revision of the manuscript. D.L., H.L. and Z.L. conducted experiments and made analysis. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the National Natural Science Foundation of China (NSF) (42161060, 41801325, 42161064), the China Post-Doctoral Science Foundation (2019M661858), the Natural Science Foundation of Jiangxi Province (20192BAB217010, 20212BAB204003), East China University of Technology Ph. D. Project (DHBK2017155) for their financial support.

Data Availability Statement: Not applicable.

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

References

1. Fang, J.Y.; Guo, Z.D.; Piao, S.L.; Chen, A.P. Terrestrial vegetation carbon sinks in China, 1981–2000. *Sci. China Earth Sci.* **2007**, *50*, 1341–1350. [CrossRef]
2. Pan, Y.; Birdsey, R.A.; Fang, J.; Houghton, R.; Kauppi, P.E.; Kurz, W.A.; Phillips, O.L.; Shvidenko, A.; Lewis, S.L.; Canadell, J.G. A large and persistent carbon sink in the world's forests. *Science* **2011**, *333*, 988. [CrossRef] [PubMed]
3. Xinliang, X.; Kerang, L. Biomass Carbon Sequestration by Planted Forests in China. *Sci. China Earth Sci.* **2010**, *20*, 289–297.
4. Cao, L.; Gao, S.; Li, P.H.; Yun, T.; Shen, X.; Ruan, H.H. Aboveground Biomass Estimation of Individual Trees in a Coastal Planted Forest Using Full-Waveform Airborne Laser Scanning Data. *Remote Sens.* **2016**, *8*, 279. [CrossRef]
5. Knapp, N.; Fischer, R.; Cazcarra-Bes, V.; Huth, A. Structure metrics to generalize biomass estimation from lidar across forest types from different continents. *Remote Sens. Environ.* **2020**, *237*, 111597. [CrossRef]
6. Kukenbrink, D.; Gardi, O.; Morsdorf, F.; Thurig, E.; Schellenberger, A.; Mathys, L. Above-ground biomass references for urban trees from terrestrial laser scanning data. *Ann. Bot.* **2021**, *128*, 709–724. [CrossRef]
7. Hu, M.; Pitkanen, T.P.; Minunno, F.; Tian, X.; Lehtonen, A.; Makela, A. A new method to estimate branch biomass from terrestrial laser scanning data by bridging tree structure models. *Ann. Bot.* **2021**, *128*, 737–751. [CrossRef]
8. Cournede, P.H.; Mathieu, A.; Houllier, F.; Arthelemy, D.B.; Reffye, P.D. Computing Competition for Light in the GREENLAB Model of Plant Growth: A Contribution to the Study of the Effects of Density on Resource Acquisition and Architectural Development. *Ann. Bot.* **2007**, *101*, 1207–1219. [CrossRef]
9. 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]
10. Hui, Z.; Jin, S.; Li, D.; Ziggah, Y.Y.; Liu, B. Individual Tree Extraction from Terrestrial LiDAR Point Clouds Based on Transfer Learning and Gaussian Mixture Model Separation. *Remote Sens.* **2021**, *13*, 223. [CrossRef]
11. Hyyppa, J.; Kelle, O.; Lehikoinen, M.; Inkinen, M. A segmentation-based method to retrieve stem volume estimates from 3-D tree height models produced by laser scanners. *IEEE Trans. Geosci. Remote Sens.* **2001**, *39*, 969–975. [CrossRef]
12. Du, S.; Lindenbergh, R.; Ledoux, H.; Stoter, J.; Nan, L. AdTree: Accurate, Detailed, and Automatic Modelling of Laser-Scanned Trees. *Remote Sens.* **2019**, *11*, 2074. [CrossRef]
13. Hui, Z.; Jin, S.; Xia, Y.; Wang, L.; Ziggah, Y.Y.; Cheng, P. Wood and leaf separation from terrestrial LiDAR point clouds based on mode points evolution. *ISPRS J. Photogramm. Remote Sens.* **2021**, *178*, 219–239. [CrossRef]
14. Calders, K.; Adams, J.; Armston, J.; Bartholomeus, H.; Bauwens, S.; Bentley, L.P.; Chave, J.; Danson, F.M.; Demol, M.; Disney, M.; et al. Terrestrial laser scanning in forest ecology: Expanding the horizon. *Remote Sens. Environ.* **2020**, *251*, 112102. [CrossRef]
15. Jin, S.; Sun, X.; Wu, F.; Su, Y.; Li, Y.; Song, S.; Xu, K.; Ma, Q.; Baret, F.; Jiang, D.; et al. Lidar sheds new light on plant phenomics for plant breeding and management: Recent advances and future prospects. *ISPRS J. Photogramm. Remote Sens.* **2021**, *171*, 202–223. [CrossRef]
16. Lau, A.; Calders, K.; Bartholomeus, H.; Martius, C.; Raumonen, P.; Herold, M.; Vicari, M.; Sukhdeo, H.; Singh, J.; Goodman, R.C. Terrestrial LiDAR: A Case Study in Guyana. *Forests* **2019**, *10*, 527. [CrossRef]
17. Guo, Q.; Su, Y.; Hu, T.; Guan, H.; Jin, S.; Zhang, J.; Zhao, X.; Xu, K.; Wei, D.; Kelly, M.; et al. Lidar Boosts 3D Ecological Observations and Modelings: A Review and Perspective. *IEEE Geosci. Remote Sens.* **2021**, *9*, 232–257. [CrossRef]
18. Kelly, M.; Di Tommaso, S. Mapping forests with Lidar provides flexible, accurate data with many uses. *Calif. Agr.* **2015**, *69*, 14–20. [CrossRef]
19. Pueschel, P.; Newnham, G.; Rock, G.; U De Lhoven, 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]
20. Shao, J.; Zhang, W.; Mellado, N.; Wang, N.; Jin, S.; Cai, S.; Luo, L.; Lejemble, T.; Yan, G. SLAM-aided forest plot mapping combining terrestrial and mobile laser scanning. *ISPRS J. Photogramm. Remote Sens.* **2020**, *163*, 214–230. [CrossRef]

21. Liang, X.; Kukko, A.; Hyyppa, J.; Lehtomaki, M.; Pyorala, J.; Yu, X.; Kaartinen, H.; Jaakkola, A.; Wang, Y. In-situ measurements from mobile platforms: An emerging approach to address the old challenges associated with forest inventories. *ISPRS J. Photogramm. Remote Sens.* **2018**, *143*, 97–107. [CrossRef]
22. Stovall, A.E.L.; Anderson-Teixeira, K.J.; Shugart, H.H. Assessing terrestrial laser scanning for developing non-destructive biomass allometry. *Forest Ecol. Manag.* **2018**, *427*, 217–229. [CrossRef]
23. Raumonen, P.; Kaasalainen, M.; Åkerblom, M.; Kaasalainen, S.; Kaartinen, H.; Vastaranta, M.; Holopainen, M.; Disney, M.; Lewis, P. Fast Automatic Precision Tree Models from Terrestrial Laser Scanner Data. *Remote Sens.* **2013**, *5*, 491–520. [CrossRef]
24. Hackenberg, J.; Morhart, C.; Sheppard, J.; Spiecker, H.; Disney, M. Highly Accurate Tree Models Derived from Terrestrial Laser Scan Data: A Method Description. *Forests* **2014**, *5*, 1069–1105. [CrossRef]
25. Hackenberg, J.; Wassenberg, M.; Spiecker, H.; Sun, D. Non Destructive Method for Biomass Prediction Combining TLS Derived Tree Volume and Wood Density. *Forests* **2015**, *6*, 1274–1300. [CrossRef]
26. Wang, Z.; Zhang, L.; Fang, T.; Mathiopoulos, P.T.; Qu, H.; Chen, D.; Wang, Y. A Structure-Aware Global Optimization Method for Reconstructing 3-D Tree Models from Terrestrial Laser Scanning Data. *IEEE Trans. Geosci. Remote Sens.* **2014**, *52*, 5653–5669. [CrossRef]
27. Bucksch, A.; Lindenbergh, R. CAMPINO-A Skeletonization Method for Point Cloud Processing. *ISPRS J. Photogramm. Remote Sens.* **2008**, *63*, 115–127. [CrossRef]
28. Bucksch, A.; Lindenbergh, R.; Menenti, M. SkelTre: Robust Skeleton Extraction from Imperfect Point Clouds. *Vis. Comput.* **2010**, *26*, 1283–1300. [CrossRef]
29. Delagrange, S.; Jauvin, C.; Rochon, P. PypeTree: A Tool for Reconstructing Tree Perennial Tissues from Point Clouds. *Sensors* **2014**, *14*, 4271–4289. [CrossRef]
30. Fan, G.; Nan, L.; Dong, Y.; Su, X.; Chen, F. AdQSM: A New Method for Estimating Above-Ground Biomass from TLS Point Clouds. *Remote Sens.* **2020**, *12*, 3089. [CrossRef]
31. Huang, H.; Wu, S.; Cohen-Or, D.; Gong, M.; Zhang, H.; Li, G.; Chen, B. L1-medial skeleton of point cloud. *ACM Trans. Graph.* **2013**, *32*, 654. [CrossRef]
32. Lu, W.; Zhang, X.; Liu, Y. L-1-medial skeleton-based 3D point cloud model retrieval. *Multimed. Tools Appl.* **2019**, *78*, 479–488. [CrossRef]
33. Wu, S.; Wen, W.; Xiao, B.; Guo, X.; Du, J.; Wang, C.; Wang, Y. An Accurate Skeleton Extraction Approach From 3D Point Clouds of Maize Plants. *Front. Plant Sci.* **2019**, *10*, 248. [CrossRef]
34. Cao, J.; Tagliasacchi, A.; Olson, M.; Hao, Z.; Su, Z. Point cloud skeletons via Laplacian based contraction. In Proceedings of the SMI 2010, Shape Modeling International Conference, Aix en Provence, France, 21–23 June 2010; pp. 187–197.
35. Su, Z.; Zhao, Y.; Zhao, C.; Guo, X.; Li, Z. Skeleton extraction for tree models. *Math. Comput. Model.* **2011**, *54*, 1115–1120. [CrossRef]
36. Wang, D.; Takoudjou, S.M.; 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]
37. Gonzalez De Tanago, J.; Lau, A.; Bartholomeus, H.; Herold, M.; Avitabile, V.; Raumonen, P.; Martius, C.; Goodman, R.C.; Disney, M.; Manuri, S.; et al. Estimation of above-ground biomass of large tropical trees with terrestrial Li-DAR. *Methods Ecol. Evol.* **2017**, *9*, 223–234. [CrossRef]
38. Wang, D.; Kankare, V.; Puttonen, E.; Hollaus, M.; Pfeifer, N. Reconstructing Stem Cross Section Shapes from Terres-trial Laser Scanning. *IEEE Trans. Geosci. Remote Sens.* **2017**, *14*, 272–276. [CrossRef]