Combining Trunk Detection With Canopy Segmentation to Delineate Single Deciduous Trees Using Airborne LiDAR Data

XIAOHU WANG1,2, YIZHUO ZHANG1, AND ZE LUO1
1College of Mechanical and Electrical Engineering, Northeast Forestry University, Harbin 150040, China
2School of Electrical Information Engineering, Hunan Institute of Technology, Hengyang 421002, China
Corresponding author: Yizhuo Zhang (zhangyz@nefu.edu.cn)

This work was supported in part by the Key Project of Hunan Provincial Science and Technology Department under Grant 2017GK2160.

ABSTRACT To minimize omission and commission errors due to the lack of adequate utilization of forest structure information, this paper presents a tree delineation approach by combining trunk detection with canopy segmentation. First, all potential tree trunk points are detected and detached from leaf-off data by analyzing the points’ vertical histogram, and the obtained points are then clustered using the method based on DBSCAN (Density-Based Spatial Clustering of Application with Noise). Meanwhile, the canopy-based segmentation is implemented using leaf-on data within the same plot. The detected trunks and delineated crown segments are then combined using the matching rules. Finally, single trees are isolated from point clouds, and tree-level structure information is estimated. The novelty of this approach lies in that the trunk detection results and the canopy segmentation results serve as mutual references for final individual tree delineation. Experimental results in a canopy-closed deciduous natural forest show that the presented method can identify 84.0% of trees, 90.7% of the identified trees are correct, and the total segmentation accuracy is 87.2%. The determination coefficient R² of tree height is 0.96, and the mean difference of tree position is 76 cm. The results imply that the presented approach has good potential for isolating single trees from airborne LiDAR point clouds and estimating tree-level structural parameters in deciduous forests.

INDEX TERMS Airborne LiDAR, canopy segmentation, deciduous forest, trunk detection.

I. INTRODUCTION

As an important part of the terrestrial ecosystem, forest and its change have an important impact on the terrestrial biosphere and the Earth’s surface changes. Accurate and convenient forest inventories are significant for the protection of the ecological environment and the well-being of human beings and wildlife [1]. In forest resource inventory campaigns, tree-level information is usually obtained by visual estimation or simple instrument measurement method, with relatively poor accuracy, which cannot meet the requirements of “precision forestry” and “digital forestry” [2]. Over the past decades, remote-sensing based inventory systems have developed rapidly aiming to reduce the difficulty and cost of forest surveys. Of current inventory systems, airborne LiDAR-based technology features prominently [3]–[5].

Due to LiDAR’s outstanding ability to map three-dimensional (3D) forest structure [6], the demand for tree-level parameters has multiplied to optimize management campaigns including forest cultivation and thinning [7]–[10]. Once a single tree has been divided from the LiDAR data, its height and crown size can be extracted directly, and the position of an individual tree can be estimated according to the crown vertex position. In combination with the allometric growth equation or regression analysis, diameter at breast height (DBH), crown base height, crown perimeter, biomass, carbon storage, and other factors can be further calculated [11].

Various methodologies have been proposed to isolate single trees from airborne LiDAR data and extract tree-level information. Earlier approaches pre-process point cloud into digital surface models (DSMs) or canopy height models (CHMs) to identify single trees [12]–[17]. DSM-based approaches search the global maximum elevation among
surface points of the point cloud and generate a vertical profile to obtain a convex hull to isolate a single tree from the point cloud [18]. CHM is obtained by calculating the difference between the DSM and the digital terrain model (DTM). CHM-based approaches use local maximum filter to detect tree apices and then utilize a marker-controlled watershed segmentation algorithm, a region growth technique or a pouring technique to depict an individual crown. The algorithm based on CHM interpolates the spatial discrete points representing the vegetation canopy surface to obtain the raster image, so there may be inherent errors and uncertainties [11]. The spatial errors caused by interpolation technology would reduce the accuracy of tree identification and tree-level information extraction [19]. Overall, the disadvantage of these methods is that the algorithm only focuses on the surface points during isolating single trees from point clouds, which leads to miss low-vegetation [20], [21].

To overcome the shortcomings of the algorithms mentioned above, more recent methods directly process the raw LiDAR point clouds. From the strategic viewpoint of the algorithms, these algorithms generally fall into two categories: top-down and bottom-up methods. The methods based on top-down utilize canopy structure information to directly search the 3D volume, aim to segment and delineate individual trees. The majority of existing approaches [11], [21]–[24] use top-down algorithms and work best for trees with a distinct top. This is usually true for coniferous forests with simple crown shape, but not for deciduous forests with asymmetric shape and complex structure. To overcome this challenge, several recent bottom-up segmentation methods have been developed. For example, Ayre et al. [25] presented a layer stacking individual tree-detection methodology, which slices the airborne LiDAR data at 1-m-height intervals, then delineates trees within the canopy of each slice, and finally combines the tree segments across layers. This approach improved single tree segmentation accuracy, however, they usually ignored information of the vertical canopy geometry during processing a 2D profile. Hamraz et al. [26] developed a stratification-enabled tree-detection method, which gradually strips the crown layer from the top to the bottom, and then segments and delineates individual tree profiles from the stripped canopy layers to detect more low-level vegetation. This method significantly improved the detection rate of understory trees (from 46% to 68%) in a deciduous forest but at the cost of introducing a considerable amount of over-segmented understory trees (increased by 15 percentage points). Wang et al. [27] developed a multi-threshold tree-detection method, which segments trees of overstory and understory using different thresholds and then combines tree-segments across canopy layers. The approach has been tested over a deciduous natural forest: the average detection rate and average overall accuracy of understory were 73.4% and 76.9%, respectively, and the overall accuracy of all trees was 82.6% [27]. It is worth noting that compared with the DSM-based method [28], this multiple threshold method can detect nearly 21% more understory trees.

Overall, compared with the CHM- or DSM-based methods, although the majority of recently developed top-down-based methods can improve the identification rate of the understory, there are still visible under- and over-segmentation errors in the final individual tree depiction results. This can be attributed two reasons: Firstly, due to the overlapping of lidar point clusters caused by complex vegetation, a cluster of trees close to each other might be merged into a single tree when performing the multi-threshold tree segmentation, however, they are delineated as multiple individual trees in the on-site inventory. Secondly, most deciduous trees are asymmetric in shape and complex in structure, with some having different crown sizes and their branches crisscrossing or stretching outward to make them look like individual trees. This scenario might result in false detecting and dividing a big tree into smaller ones. In a word, the top-down segmentation strategies have an inherent defect, they only focus on the canopy information, and the segmentation accuracy mainly depends on the accurate delineation of tree canopies. However, in practice, it is not a trivial problem to identify and characterize understory from natural broad-leaved forests with complex structures.

On the other hand, a few recent studies try to use the bottom-up segmentation strategy to achieve higher tree detection accuracy. The bottom-up methods extract and segment tree trunks from the point cloud, and then delineate individual crowns with trunks as seed points. Reitberger et al. [29] utilized the three-dimensional line adjustment technology based on random sample consensus (RANSAC) to identify trunks and a normalized cut to isolate individual trees from point clouds. Lu et al. [30] presented a bottom-up approach based on lidar point-intensity and 3D structure to detect tree trunks. It is worth noting that Reitberger et al. [29] and Lu et al. [30] only evaluated the accuracy of trunk identification in the sample plots, but did not extract tree-level information and evaluate it (e.g. tree position, height, apex, and crown size). To extract more tree-level information, a two-step, bottom-up tree segmentation approach was proposed by Shendryk et al. [31], in which tree stems are detected using the conditional Euclidean distance clustering method, individual tree canopies are then isolated utilizing a method based on random walks. The results showed that the determination coefficients ($R^2$) of the measured and lidar-estimated tree heights and crown widths were 0.92 and 0.41, respectively [31]. It is worth noting that the $R^2$ of crown width shows a weak correlation. Chen et al. [32] presented an adaptive mean shift-based clustering method to isolate single trees from deciduous forests, which utilizes detected trunks to adjust the kernel bandwidth of the clustering algorithm. However, the segmentation performance depends on the results of trunk detection, and not all trunks can be correctly identified. In general, the disadvantages of the bottom-up methods could be attributed to the use of tree trunk information as seed points for clustering tree segmentation, because selecting different points as seeds might lead to significant differences in the crown description. In addition, the density of the lidar point is
an important factor affecting the accuracy of the identification and depiction of single trees, which should be considered before the remote sensing-based survey [5], [31].

Generally speaking, the top-down segmentation method usually focuses on the canopy segmentation. Due to the lack of tree trunk information, it might lead to under-segmentation and commission errors of undergrowth vegetation; on the contrary, the bottom-up segmentation method often focuses on tree trunk detection, whereas the tree segmentation method with tree trunk information as seed points is greatly affected by noise points. From the perspective of LiDAR data characteristics, leaf-on data obtained in the leafy season can reflect the structural characteristics of the canopy, which may be suitable for the canopy-based segmentation algorithm. However, leaf-off data is obtained in the leafless season without the influence of leaf occlusion, which contains abundant structural information of undergrowth and may be suitable for trunk detection. Therefore, in this study, we try to combine trunk detection with canopy segmentation to delineate individual deciduous trees using both leaf-off and leaf-on data, with the aim to improve the accuracy of tree delineation and tree-level parameter extraction. More specifically, the purposes of this study are to (1) propose an approach to isolate single trees from airborne LiDAR point clouds by detecting trunks from leaf-off data and segmenting tree crowns from leaf-on data; (2) investigate the feasibility of combining detected trunk information with isolated tree crown segments; (3) assess the new method with respect to the metric of recall, precision, and tree-level parameter extraction accuracy.

II. STUDY AREA AND DATA

A. STUDY SITE AND FIELD DATA

The data of this study are from the Robinson forest (RF, coordinates: 37°28'23"N 83°08'36"W) of the University of Kentucky, which is located in the rugged east of the Cumberland Plateau in Southeast Kentucky. The terrain of RF is rugged and complex, rich in narrow ridges and valleys, and the elevation of the ridges ranges from 252 to 503 meters. The total area of RF is about 7740 ha. It is a canopy-closed deciduous forest with about 80 tree species, about 2.5 million trees (±13.5%), of which more than 60% are undergrowth trees. The average coverage rate of the whole forest land is 93%, and the coverage rate of most areas is higher than 97%. After commercial logging in the 1920s, coverage is recent as low as 63% in a few areas. RF is now protected from commercial logging and mining. In the summer of 2013, 271 regularly distributed circular plots of 0.04 ha size were surveyed in the field. 1.2 m × 1.2 m plywood boards were installed in the plots and painted white to increase reflectivity. The exact positions of plywood boards were recorded to accurately map the center of the plot. In each plot, all trees with DBH greater than 12.5 cm were recorded tree-level parameters, such as tree species, tree height, DBH, crown category (dominant, co-dominant, intermediate, overtopped), tree status (alive and dead) and trunk category (single and multiple). Also, the horizontal distance and azimuth from the center of the plot to each tree at breast height were recorded to generate a trunk map. Each plot was also acquired with parameters such as slope, aspect, and slope location [26], [27]. Table 1 shows the plot-level parameters of the 271 plots in RF.

B. LIDAR CAMPAIGN

The LiDAR data are composed of two parts: one was collected in the summer of 2013 during the leaf-on season, and the LiDAR point density was about 25 pt/m². It was ordered by the University of Kentucky Department of Forestry, to obtain detailed information on vegetation structure. Another part of the data was collected in the spring of 2013 during the leaf-off season, and the LiDAR point density was about 2 pt/m².

| Parameters | Leaf-on data | Leaf-off data |
|------------|--------------|---------------|
| Sensor name | Leica ALS60   | Leica ALS60   |
| Date of Acquisition | May 28-30, 2013 | April 23, 2013 |
| Pulse Repetition Rate | 200 KHz | 200 KHz |
| Field of View | 40° | 40° |
| Average Flight Speed | 105 knots | 105 knots |
| Average Flight Elevation above Ground | 214 m | 3096 m |
| Swath Width | 142.7 m | 2253.7 m |
| Swath Overlap | 50% | 50% |
| Usable Center Portion of Swath | 95% | 90% |
| Point Density | ~ 25 pt/m² | ~ 2 pt/m² |
| Average Footprint | 0.15 m | 0.6 m |
| Nominal Post Spacing | 0.2 m | 0.8 m |
It was obtained by the Kentucky Division of Geographic Information for obtaining terrain information as a part of the state’s elevation data collection plan. The two data sets were acquired by the same survey company using the same LiDAR system \cite{26}, \cite{27}. The parameters statistics of the LiDAR system are shown in Table 2.

III. METHODOLOGY

TerraScan software was adopted to preprocess the two original LiDAR data sets (namely leaf-on and leaf-off dataset) and classify LiDAR points into ground and vegetation points. LASTools software was utilized to generate a single airborne laser scanning (ALS) dataset file. Then, the DTM with 1-meter resolution was generated, and the mean and nearest neighbor algorithm were utilized for interpolation and padding respectively. The following pre-processing routine contains five steps: (1) uniformize lidar point density to eliminate the uneven distribution of points due to the scanning pattern (e.g., triangle and sine pattern), (2) generate a raster with a resolution value of NPS (nominal point spacing), (3) search the highest points within each raster unit, and the searched point set is used as lidar surface points to filter the LiDAR data, (4) calculate the heights above ground for all lidar surface points utilizing DTM, and (5) a Gaussian filter is used to smooth the lidar surface points to weaken the subtle changes of canopy vegetation height. In this study, the standard deviation and radius of the Gaussian filter are set to NPS and 3 \times NPS respectively. As a result, important vegetation patterns were maintained.

After this pre-processing routine, the canopy-based segmentation method was implemented to isolate individual tree canopies from the leaf-on point cloud containing rich canopy information (Section A). Tree trunks were detected and isolated using the leaf-off point cloud belonging to the same plot (Section B). After that, the LiDAR-derived crown segments and detected trunks were matched, the final LiDAR-derived trees were then delineated, and the tree-level information was estimated (Section C). Finally, the effectiveness of the method was evaluated (Section D). Fig. 1 shows the methodology framework of this study.

A. CANOPY-BASED SEGMENTATION ALGORITHM

In this step, the leaf-on point cloud is segmented utilizing the multi-threshold segmentation method introduced by Wang et al. \cite{27}, the canopy-based segmentation routine consists of the following three steps: (1) stratify point cloud according to lidar returns and code the stratified canopy layers; (2) Set segmenting threshold values (the ideal values for the first, second and third layers here are 1.2 \times NPS, 1.5 \times NPS and 2.0 \times NPS, respectively); and (3) implement tree segmentation within each stratified canopy layers using different thresholds. Fig. 2 shows the canopy-based segmentation method. For more details about this method, see [27].

The object of this method is the point clouds representing the tree canopies, and we segmented the individual trees by analyzing the vegetation canopy structure represented by the point clouds. This multi-threshold segmentation method utilizes different segmentation thresholds for upper vegetation canopy and lower vegetation canopy, which can detect more low trees and improve the detection rate, but at the cost of the accuracy of tree detection. There are two reasons for this: first, some large trees have large branches spread out, which are recognized as a single tree (commission errors). Second, some small trees are located under the canopy of a big tree and are identified as part of the big tree (omission errors). In general, only utilizing a canopy-based tree segmentation would result in commission errors, or omission errors, or both, for the reason that a canopy-based tree delineation mainly depends on a profiler generation of tree crowns, but not on a tree trunk detection. Therefore, trunk detection will be helpful to improve the overall segmentation accuracy during tree segmentation.

Meanwhile, to extract tree-level information accurately, it is necessary not only to delineate tree crowns accurately but also to isolate tree trunks accurately. Zhao et al. \cite{5} pointed out that point density is the most important parameter of LiDAR data for treetop detection and tree height estimation. Leaf-on data with high density (~25 pt/m² in this study) can meet this requirement, but because of the shading of overlapping leaves, the number of lidar pulses reaching understory is reduced, so leaf-on data are lack of the tree trunk information. In contrast, leaf-off data, although low in point density (~2 pt/m²), are collected during the leaf-off season, and most lidar pulses can reach tree trunks and characterize their three-dimensional information. In this study, the leaf-on LiDAR data are used to segment and delineate tree crowns, and the leaf-off LiDAR data are used to detect and identify tree trunks.

B. TREE TRUNKS DETECTION

1) DBSCAN CLUSTERING THEORY

Different from canopy-based tree segmentation, noise points (points from the understory shrubs) contained in the LiDAR point clusters have a great impact on the accuracy of trunk identification. DBSCAN \cite{33} clustering technology is a kind of density-based clustering algorithm, which can effectively weaken the impact of noise on clustering accuracy, providing a promising method for tree trunk detection.

To introduce the DBSCAN clustering algorithm, given data set \( D = \{x_1, x_2, \ldots, x_m\} \), the following concepts are defined:

- \( \text{Eps} \)-neighborhood: the \( \text{Eps} \)-neighborhood of a point \( x_j \), denoted by \( N_{\text{Eps}}(x_j) \), is defined by \( N_{\text{Eps}}(x_j) = \{x_i \in D | \text{dist}(x_i, x_j) \leq \text{Eps}\} \).

- Core object: If \( N_{\text{Eps}}(x_j) \) contains at least a minimum number (MinPts) of points, namely \( |N_{\text{Eps}}(x_j)| \geq \text{MinPts} \), then \( x_j \) is a core object;

- Directly density-reachable: a point \( x_j \) is directly density-reachable form a point \( x_i \) wrt. Eps, MinPts if

\[
\begin{align*}
1) & \ x_i \in N_{\text{Eps}}(x_j) \\
2) & x_i \text{ is a core object.}
\end{align*}
\]
Density-reaching: a point $x_j$ is *density-reachable* from a point $x_i$ if there is a chain of points $q_1, \ldots, q_n$, $q_1 = x_i$, $q_n = x_j$ such that $q_{i+1}$ is directly density-reachable from $q_i$.

Density-connected: a point $x_j$ is *density-connected* to a point $x_i$ if there is a point $x_k$ such that both, $x_j$ and $x_i$ are density-reachable from $x_k$.

It is important to note that a set of “neighborhood” parameters ($Eps$, $MinPts$) is required for DBSCAN, need not be defined simultaneously. $Eps$ denotes the radius of clustering, $MinPts$ denotes the minimum amount of points contained within the circle, namely, the density threshold of clustering. $Eps$ can be obtained by the spatial size of the clustering object and the empirical value of $MinPts$. The calculation formulas are as...
follows:

\[
Eps = \sqrt{\frac{T \cdot \text{MinPts} \cdot \Gamma [(1/2) \cdot n + 1]}{m \sqrt{\pi n}}} \quad (1)
\]

\[
T = \prod_{i=1}^{n} \left( \max \left( X_i \right) - \min \left( X_i \right) \right) \quad (2)
\]

where \( m \) and \( n \) denote the amount and dimension of points respectively, and \( T \) denotes the size of the clustering space composed of \( m \) points. Fig. 3 shows the Flowchart of the DBSCAN clustering algorithm applied by our approach.

2) TRUNK ISOLATION AND IDENTIFICATION

The trunk isolation and identification in our methodology consist of three steps.

First, separate all latent canopy lidar point cloud from the leaf-off data within a plot and divide out the trunk part of the point cloud (Fig. 4). This target is obtained by analyzing the distribution of lidar points of point cloud within the plot. The idea of this separation technique is similar to that proposed in [29] and [34]. This step is implemented by performing the following routines: (1) split the LiDAR point cloud within a plot into \( N_l \) layers with a height of 3 meters,

(2) count the lidar points \( n_l \) of each layer within each plot and calculate the percentage of \( n_l \) in the number of points, (3) form the distribution histogram of lidar points at various height-level, (4) search the layers where the percentage of points exceeds a preset threshold value \( h_{base} \) and, finally (5) define the highest layer \( l_x \) among the searched out layers as the dividing plane. The points below \( l_x \) are the potential trunk points and can result from one or several trunks.

Secondly, the divided trunk points are clustered with respect to the spatial distance to obtain the estimated amount and position of trunks within a plot, and the points are assigned to the trunk point clusters (Fig. 5). To eliminate noise points from the understory or isolated branches, and to depict trunks with different shapes (e.g., curved, twisted), DBSCAN-based clustering method is adopted to detect and isolate tree trunks. The points are assigned to the estimated trunk point clusters without prior knowledge of the trunk’s shape and amount. Before the clustering routine, all the separated trunk points are projected onto the \( x-y \) plane, and then DBSCAN-based clustering is implemented in the two-dimensional space composed of projected points. The initial values of these two parameters (\( Eps, \text{MinPts} \)) are calculated by (1) and (2), and then the optimal values for the

*FIGURE 3. Flowchart of the DBSCAN clustering algorithm.*
plot are obtained by comparison of multiple iterations. In our study case, the ideal values are 2 and 5, respectively.

Finally, the identified tree trunk point clusters are post-processed by excluding the following cases: (1) the height of the lowest point of a cluster is greater than 4 meters (would be considered as a part of the crown point segments); (2) the height of the highest point of the cluster is less than 2 meters (would be considered as a shrub); (3) the interquartile range (IQR) of the cluster is less than 1 meter (would be considered as fallen trees).

**C. MATCHING TREE CROWN SEGMENTS AND TRUNKS**

After trunk detection and crown segmentation, the trunk information and crown segments are combined and referred to each other to delineate the final single trees. On this basis, we can make final decisions to segment and depict...
single trees. The purpose of this stage is to improve the total precision of tree segmentation, especially for the forest with a complex structure, in which most trees overlap.

Given the number of canopy layers \( i (i = 1, 2, 3) \), the number of detected tree trunks is \( j (j = 0, 1, 2, \ldots) \). Project segmented crown point clusters within each layer onto the \( x-y \) plane, examine the extracted trunk information taking crown point segments into account. Depending on the number of detected tree trunks, the matching between tree crown segments and trunks contains the following cases, as shown in Fig. 6:

1. No trunk is contained within a crown segment, such as \( R_3 \) and \( R_4 \). This indicates that the crown segment would be treated as a branch and combined into the adjacent tree nearest to it.

2. One trunk is contained within a crown segment, such as \( R_1, R_2, R_5, R_6, \) and \( R_7 \). This case implies that the trunk matches the crown segment within the given canopy layer. As a candidate matching pair, whether their point clusters are combined to outline the entire structure of a single tree needs to be determined in further matching steps.

3. Several trunks are contained within a crown segment, such as \( R_8 \). This case means that there are some small trees covered by a large tree.

In a few scenarios, no crown segment is detected in the area directly above the trunk. This case implies that there are no branches belong to it, and is identified as a withered tree without further study.

Thus, the cross-combination rule presented by Wang et al. [27] is improved, as shown in Fig. 6(a), where after trunk detection, there are two trunks (trunk 2 and trunk 3) contained in crown segments \( R_8 \). According to the matching technique proposed in this paper (Fig. 7), the crown segments \( R_2 \) and \( R_6 \) are combined to form a single tree, which is no longer merged into \( R_8 \) as branches. Based on the above description, the improved connection rule of crown segment nodes (i.e., the improvement of the so-called “connected growth tree” (CGT) [27]) shown in Fig. 6(b). It demonstrates the fusion mechanism of nodes in the canopy at different height levels, where each node represents a crown segment.

After selecting the set of pairs with the shortest distance where the trunks or crown segments appear no more than once, we regard that the trunks match the crown segments. We select the matched segments as root nodes (such as \( R_5, R_6, \) and \( R_8 \) in Fig. 6), which are not merged into other nodes. The remaining crown-segment point clusters constitute the
Fig. 8 shows a comparison between the multi-threshold approach and the proposed approach of combining trunk detection with canopy segmentation. It is apparent visually that the undivided tree (marked in green in Fig. 8(a)) is accurately isolated and delineated using the method of combination of trunk detection and canopy segmentation.

The individual trees were delineated after matching of crown point clusters and trunk point clusters. Tree-level parameters such as tree height, tree amount, crown size, and trunk position within the plot can be estimated from the isolated tree [27]. The tallest lidar point amongst the LiDAR-derived tree is identified as the treetop, and its elevation is considered as the tree height. The tree crown perimeter is defined as the length of the boundary line projected by the tree crown onto the x-y plane. After the trunk detection, the stationary points on the x-y plane of the cluster are identified as the approximate position of the relevant tree clusters. Since our primary goal was to present a method for isolating single trees from the LiDAR data and to evaluate the effectiveness of the method, we focused on two tree-level parameters with field measurements, namely the height and position of the tree.

D. APPROACH ASSESSMENT
To evaluate the accuracy of the 3D segmentation of individual trees, we adopted the semi-automatic algorithm presented by Zhao et al. [5] to pair the estimated trees (or lidar-derived trees) with the reference trees (or field-measured trees).

Firstly, for the two datasets (namely, the estimated trees and the reference trees), the distance between two 3D space points representing the location information of the tree was calculated as follows:

\[
D = \sqrt{\frac{(x_1 - x_2)^2 + (y_1 - y_2)^2}{\text{planimetric distance}}} + w \cdot \frac{(z_1 - z_2)^2}{\text{height diff}}
\]  

(3)
where, \((x_1, y_1)\) and \(z_1\) denote trunk position and tree height of the estimated tree, respectively; \((x_2, y_2)\) and \(z_2\) denote on-site measured trunk position and tree height, respectively; \(w\) represents the weighting coefficient of height difference and horizontal distance (the ideal value of \(w\) was 0.5 here.).

Secondly, we matched a tree’s position in one set to another only when the two three-dimensional spatial points representing the location information of the tree were closest to each other. Using (3) and the above pairing algorithm, a preparatory list of matched trees in two data sets is automatically generated.

Finally, utilizing the Hungarian assignment method, we selected the set of pairs with the shortest distance and no more than once the occurrence of the estimated or reference tree location and regarded the dataset as the matched trees [27].

For a tree-level evaluation, if an actual tree is isolated from the LiDAR data, it is called matched tree (MT); if the tree is not isolated but allocated as a branch to an adjacent tree, it is called an omission error (OE) tree; if a branch of a tree is separated from the point cloud as a separate tree, the split tree is called a commission error (CE) tree. The number of OE trees and CE trees indicates under- and over-segmentation, respectively. To assess the accuracy of the method, we used the following formulas to calculate the identification rate of the tree (namely, recall rate (Re)), the correctness of the identified trees (namely, precision (Pr)), and the total identification accuracy (namely, F-score (F));

\[
Re = \frac{MT}{MT + OE} \quad (4) \\
Pr = \frac{MT}{MT + CE} \quad (5) \\
F = \frac{(2 \cdot Re \cdot Pr)}{(Re + Pr)} \quad (6)
\]

IV. RESULTS

A. ACCURACY OF TREE SEGMENTATION

The statistics of detection results of trees with DBH larger than 12.5 cm in the 271 test plots are shown in Table 3. Compared with the multi-threshold and DSM-based methods, the proposed approach of the combination of trunk detection and canopy segmentation was able to identify more trees and had the highest identification rate of 84 percent. More importantly, with the improvement of the identification rate, the number of the correctly-identified trees also increased, which resulted in an improvement of the accuracy rate of identified trees, with Pr up to 90.7 percent. The overall segmentation accuracy was improved to 87.2 percent by taking the tree identification rate and accuracy of identified trees into account.

To assess the influence of the detected trunks as auxiliary information for segmenting canopies at different height-level, we calculated the segmentation scores of the understory and overstory by three methods, as shown in Table 4. Compared with the understory, the segmentation scores of Re, Pr, and F of the overstory are higher, which means that the segmentation methods have better effects for the overstory trees. It is worth noting that compared with the overstory trees, the overall segmentation accuracy F of the understory trees is significantly improved by using the method proposed in this paper (4.8 percentage points vs. 2.2 percentage points), which indicates that the proposed method has a more obvious improvement on the low-vegetation segmentation. In particular, compared with the DSM-based approach, the Pr score of the multi-threshold approach decreases slightly with the increase of Re score (from 88.7 percent to 86.7 percent), whereas the Pr score of the proposed method increases with the increase of Re score (from 88.7 percent to 90.3 percent), which indicates that detected trunks as auxiliary information can effectively improve the accuracy of tree identification, especially for the understory.

B. ACCURACY OF TREE-LEVEL PARAMETERS

We evaluated the tree-level structural parameters of correctly identified individual trees in the 271 plots. Among the extracted parameters, only the tree height and position have the field measured data. The field measurement of tree height was only carried out in 84 plots, a total of 1240 trees with...
the DBH larger than 12.5 cm. In the 84 plots, 1159, 1135, and 897 trees were correctly identified using the proposed method of combination of trunk detection and canopy segmentation, the multi-threshold, and the DSM-based method, respectively. The scatter diagrams of the height relationship between the estimated and the reference trees are shown in Fig. 9. Among the three methods, the method proposed in this paper has the highest score of the determination coefficient $R^2$ of 0.96.

After collecting the trunk-map of estimated trees, the distribution of distances between the LiDAR-derived and the actual tree location for all correctly-identified trees was obtained. As above-mentioned, both the multi-threshold approach and the DSM-based approach utilize LiDAR surface points of canopies (or each canopy at different height levels) to identify the profiles of segmented trees, which are canopy-based segmentation approaches. It is worth noting that, for these canopy-based tree-segmentation approaches, due to lack of trunk information, the LiDAR-estimated apexes are usually regarded as the approximate value of trunk’s locations during the analysis of location relationship of estimated and reference trees. For the sake of fairness, when analyzing the distance distribution of estimated trees and reference trees in this paper, two kinds of data were used as the positions of the estimated trees, i.e. trunks and apexes. The distribution histogram of the horizontal distance between the estimated and the reference trees of the three methods is shown in Fig. 10. Taking tree trunk locations and apex locations as the trunk-map respectively (i.e., M-trunk and M-apex), the mean differences of our presented approach of the combination of trunk detection and canopy segmentation were 71 cm and 76 cm, respectively. However, taking tree apex locations as the trunk-map, the multi-threshold approach and the DSM-based segmentation were 82 cm and 105 cm, respectively.

V. DISCUSSION

In this study, we introduced and evaluated a new method to separate individual trees from airborne LiDAR data and extract tree-level parameters. We focused on evaluating the benefits of trunk detection combined with tree crown segmentation in the model for the identification of single trees. The presented method of combination of trunk detection and canopy segmentation uses tree trunk information as a reference for tree segmentation. It improves the identification rate as well as the correctness of the identified trees.
Due to the development of forest inventory system based on remote sensing technology and the requirement of "precision forestry", an increasing number of forestry research focuses on the delineation of individual trees and parameter extraction at the tree-level [24]–[26], [29]–[32], [35]–[38]. For instance, Chen et al. [32] improved the mean shift clustering algorithm by using trunks as auxiliary information to divide single trees from lidar data and evaluated the extracted tree-level information. However, the performance of trunk identification depends upon LiDAR point density and canopy style [32]. If the point density is high, and the canopy is open, with few leaves sheltering, most of the lidar pulses can reach the trunks, then the trunk points can be clearly gathered and identified. On the contrary, only a small number of lidar pulses reach the understory of the forest, and when the trunk points cannot be separated from the canopy effectively, the identification rate of the trunk decreases significantly, resulting in the low efficiency of the tree segmentation algorithm. Many tree segmentation methods try to use leaf-off data to improve the precision of tree delineation. Duncanson et al. [36] used the watershed-based tree segmentation method to isolate single trees from the leaf-off LiDAR data. The results show that compared with the leaf-on data, the leaf-off data are more conducive to improving the segmentation accuracy of understory trees, but it also reduces the estimation accuracy of tree-level structural parameters. The improvement of the identification rate of understory may be due to the increase of lidar points characterizing the structure of undergrowth vegetation. However, the increase of lidar pulse penetration will also lead to an increase in the canopy gap, which results in a decrease in the accuracy canopy structure description. For forest inventory campaigns focused on tree-level structural parameters, this trade-off should be considered before LiDAR data collecting activities. In general, the leaf-on season appears to be the more suitable LiDAR data-collecting time to segment tree crown for accurately extracting tree crown parameters, such as tree height, apex, and crown perimeter [29]. In contrast, the leaf-off season appears to be the more suitable LiDAR data-collecting time to extract tree trunk information for obtaining the accurate tree positions (Figs. 4(a) and 4(b)). Our results indicate that the combinational use of leaf-on data (to segment tree crowns) and leaf-off data (to detect trunks) performance better than using single data (Tables 3 and 4).

The second merit, as well as a novelty of our approach, is the strategy of a combination of trunk detection and canopy segmentation. There are many existing methods of canopy-based segmentation [11], [21]–[28], and trunk detection [29]–[32], and so far, few works of literature have studied the combination of the two methods. We use the detected tree trunks to assist in the ultimate individual trees delineation, that is, the trunks extracted from leaf-off data and crown points segmented from leaf-on data would serve as cross-references to decrease omission and commission errors (Fig. 8). Different from the existing clustering algorithm based on seed points [11], [30], [32], the detected tree trunk is only used as a reference for crown segment merging not as a seed point to divide the tree crown, which avoids the risk of generating wrong seed points due to the influence of understory noise on trunk detection in deciduous forests.

The third merit is that the estimation accuracy of the position parameters of the LiDAR-derived trees is significantly improved (Fig. 10). In many scenarios, trees have more or less inclined angles along the vertical direction, therefore, it’s not surprising that compared with the tree apex locations as the trunk-map, taking tree trunk locations as the trunk-map must be more precise.

However, our algorithm has some limitations. Firstly, although we utilize the leaf-off data to detect trunks, which is good for separating trunk points from crown points and detecting trunks, there are not all tree trunks being correctly identified due to the low point-density of the leaf-off data (~2 pts/m²). The trunk detection works successfully within the plots, where there are sufficient lidar pulses to reach the trunk section and trunk points can be effectively clustered. However, in the event that the reflections of the lidar pulses reached the trunk region are infrequent, or the understory trees are overlapped seriously, it could not effectively detect the trunks. In these scenarios, due to the lack of trunk as a reference, the matching rule of trunk and crown segments proposed in this study changes into the cross-layer combination rule of crown segments presented by Wang et al. [27]. As a result, the accuracy of detected trees is not improved, and the performance of segmentation is consistent with the previous method. One potential solution is to collect leaf-off data with a high point-density. The experimental results suggested that the detection rate cannot be improved significantly when the nominal point-density of the LiDAR point cloud is greater than 10 pts/m² [29].

Secondly, compared with the DSM-based and the multi-threshold approach, this method improves the detection rate and accuracy of detected trees, but at the same time reduces the computational efficiency, which is a trade-off to be considered. Although the computational time and whole efficiency were not recorded as the research objectives of this paper, we note that the computational time of the method of combination of trunk detection and canopy segmentation increases significantly with the increase of the size or amount of plots. Our research suggests that although the proposed method does have better performance than the DSM-based method and the multi-threshold method in the tested forest types, it may be most suitable for the case of canopy-closed deciduous forests (with leaf-on and leaf-off situations), or in the case of small plot area requiring high precision.

In addition, limitations resulting from the cost associated with the data collection should also be considered, as it is not possible to always acquire multi-temporal data, especially for small-scale or independent studies.

VI. CONCLUSIONS

Airborne LiDAR can play a crucial role in forest inventory based on remote sensing technology. In order to achieve this
goal, it is very necessary and important to accurately implement tree segmentation and extract tree-level structure information from LiDAR point clouds. In this study, we utilized leaf-on and leaf-off data to develop a new method to segment and delineate single trees by combining the information of the trunk and crown segment. The implementation of our method consists of two routines: first, detect and identify tree trunks from leaf-off LiDAR data, while implement tree segmentation for tree canopy using the multi-threshold method; and next, match the detected trunks with the segmented crown segments. The accuracy evaluation metrics we used show that the proposed approach works well in the RF, a canopy-closed deciduous natural forest. Improvements were noted in both the overall accuracy of tree segmentation, as well as the accuracy of tree-level parameters. The method of combination of trunk detection and canopy segmentation seems to be especially suitable for deciduous forests with leaf-on and leaf-off data. We believe that the presented algorithm is conducive to the rapid developments of tree-level parameter extraction methods in the use of airborne LiDAR data, all of which are expected to achieve the requirements of "precision forestry" and "digital forestry".

ACKNOWLEDGMENT

The authors would like to appreciate the Department of Forestry at the University of Kentucky for providing data for this research. They declare no conflict of interest.

REFERENCES

[1] J. Liu, J. Shen, R. Zhao, and S. Xu, “Extraction of individual tree crowns from airborne LiDAR data in human settlements,” Math. Comput. Model., vol. 58, nos. 3–4, pp. 524–535, Aug. 2013.
[2] H. Lee, K. C. Slatton, B. E. Roth, and W. P. Cropper, “Adaptive clustering of airborne LiDAR data to segment individual tree crowns in managed pine forests,” Int. J. Remote Sens., vol. 31, no. 1, pp. 117–139, Jan. 2010.
[3] C. Babcock, A. O. Finley, J. B. Bradford, R. Kolka, R. Birdsey, and M. G. Ryan, “LiDAR-based prediction of forest biomass using hierarchical models with spatially varying coefficients,” Remote Sens. Environ., vol. 169, pp. 113–127, Nov. 2015.
[4] R. O. Dubayah and J. B. Drake, “LiDAR remote sensing for forestry,” J. Forestry, vol. 98, no. 6, pp. 44–46, Jun. 2000.
[5] K. Zhao, J. C. Suarez, M. Garcia, T. Hu, C. Wang, and A. Londo, “Utility of multitemporal lidar for forest and carbon monitoring: Tree growth, biomass dynamics, and carbon flux,” Remote Sens. Environ., vol. 204, pp. 883–897, Jan. 2018.
[6] K. T. Vierling, L. A. Vierling, W. A. Gould, S. Martinuzzi, and R. M. Clawes, “Lidar: Shedding light on habitat characterization and modeling,” Frontiers Ecol. Environ., vol. 6, no. 2, pp. 90–98, Mar. 2008.
[7] L. Duncanson, R. Dubayah, G. Hutt, N. Pinto, B. Cook, and A. Swatantran, “How important is individual tree information for biomass modeling and mapping?” in Proc. AGU Fall Meeting Abstracts, San Francisco, CA, USA, 2012, p. 0353.
[8] C. Chen, M. Wang, B. Chang, and Y. Li, “Multi-level interpolation-based filter for airborne LiDAR point clouds in forested areas,” IEEE Access, vol. 8, pp. 41000–41012, Feb. 2020.
[9] Z. Q. Xu, W. Z. Li, Y. Y. Li, X. Shen, and H. H. Ruan, “Estimation of secondary forest parameters by integrating image and point cloud-based metrics acquired from unmanned aerial vehicle,” J. Appl. Rem. Sens., vol. 14, no. 2, 022204, Sep. 2019.
[10] M. A. Wulder, J. C. White, R. F. Nelson, E. Næsset, H. O. Ørka, N. C. Coops, T. Hikker, C. W. Bater, and T. Gobakken, “Lidar sampling for large-area forest characterization: A review,” Remote Sens. Environ., vol. 121, pp. 196–209, Jun. 2012.
[11] W. Li, Q. Guo, M. K. Jakubowski, and M. Kelly, “A new method for segmenting individual trees from the lidar point cloud,” Photogramm. Eng. Remote Sens., vol. 78, no. 1, pp. 75–84, Jan. 2012.
[12] Q. Chen, D. Baldochhi, P. Gong, and M. Kelly, “Isolating individual trees in a savanna woodland using small footprint lidar data,” Photogramm. Eng. Remote Sens., vol. 72, no. 8, pp. 932–932, Aug. 2006.
[13] L. Jing, B. Hu, J. Li, and T. Noland, “Automated delineation of individual tree crowns from LiDAR data by multi-scale analysis and segmentation,” Photogramm. Eng. Remote Sens., vol. 78, no. 12, pp. 1275–1284, Dec. 2012.
[14] B. Koch, U. Heyder, and H. Weinacker, “Detection of individual tree crowns in airborne LiDAR data,” Photogramm. Eng. Remote Sens., vol. 72, no. 4, pp. 357–363, Apr. 2006.
[15] D.-A. Kwak, W.-K. Lee, J.-H. Lee, G. S. Biging, and P. Gong, “Detection of individual trees and estimation of tree height using LiDAR data,” J. Forest Res., vol. 12, no. 6, pp. 425–434, Dec. 2007.
[16] S. Popescu and R. Wynne, “Seeing the trees in the forest: Using LiDAR and multi-spectral data fusion with local filtering and variable window size for estimating tree height,” Photogramm. Eng. Remote Sens., vol. 70, no. 5, pp. 589–604, May 2004.
[17] C. Véga and S. Durrieu, “Multi-level filtering segmentation to measure individual tree parameters based on LiDAR data: Application to a mountainous forest with heterogeneous stands,” Int. J. Appl. Earth Observ. Geoinf., vol. 13, no. 4, pp. 645–656, Aug. 2011.
[18] H. Kaartinen, J. Hyppä, X. Yu, M. Vastaranta, H. Hyppä, A. Kakko, M. Holopainen, C. Heipke, M. Hirschmugl, F. Morfford, E. Næsset, J. Solberg, P. Popescu, S. Solberg, R. M. Wolf, and J.-C. Wu, “An international comparison of individual tree detection and extraction using airborne laser scanning,” Remote Sens., vol. 4, no. 4, pp. 950–974, Dec. 2012.
[19] Q. Guo, W. Li, H. Yu, and O. Alvarez, “Effects of topographic variability and lidar sampling density on several DEM interpolation methods,” Photogramm. Eng. Remote Sens., vol. 76, no. 6, pp. 701–712, Jun. 2010.
[20] G. Shao and K. M. Reynolds, Computer Applications in Sustainable Forest Management: Including Perspectives on Collaboration and Integration. Berlin, Germany: Springer, 2006, pp. 43–66.
[21] Y. Wang, H. Weinacker, and B. Koch, “A lidar point cloud based procedure for vertical canopy structure analysis and 3D single tree modelling in forest,” Sensors, vol. 8, no. 6, pp. 3938–3951, 2008.
[22] C. Vega, A. Harmouni, S. El Mokhtari, J. Morel, J. Bock, J.-P. Renaud, M. Bouvier, and S. Durrieu, “PTrees: A point-based approach to forest tree extraction from lidar data,” Int. J. Appl. Earth Observ. Geoinf., vol. 33, pp. 98–108, Dec. 2014.
[23] T. Lahivaara, A. Seppanen, J. P. Kaipio, J. Vauhkonen, L. Korhonen, T. Tokola, and M. Maltamo, “Bayesian approach to tree detection based on airborne laser scanning data,” IEEE Trans. Geosci. Remote Sens., vol. 52, no. 5, pp. 2690–2699, May 2014.
[24] X. Hu, W. Chen, and W. Xu, “Adaptive mean shift-based identification of individual trees using airborne LiDAR data,” Remote Sens., vol. 9, no. 2, p. 148, 2017.
[25] E. Ayrey, S. Fraver, J. A. Kershaw, L. S. Kenefic, D. Hayes, A. R. Weiskittel, and B. E. Roth, “Layer stacking: A novel algorithm for individual forest tree segmentation from LiDAR point clouds,” Can. J. Remote Sens., vol. 43, no. 1, pp. 16–27, Jan. 2017.
[26] H. Hamraz, M. A. Contreras, and J. Zhang, “Vertical stratification of forest canopy for segmentation of understory trees within small-footprint airborne LiDAR point clouds,” ISPRS J. Photogramm. Remote Sens., vol. 130, pp. 385–392, Aug. 2017.
[27] X.-H. Wang, Y.-Z. Zhang, and M.-M. Xu, “A multi-threshold segmentation for tree-level parameter extraction in a deciduous forest using small-footprint airborne LiDAR data,” Remote Sens., vol. 11, no. 18, p. 2109, 2019.
[28] H. Hamraz, M. A. Contreras, and J. Zhang, “A robust approach for tree segmentation in deciduous forests using small-footprint airborne LiDAR data,” Int. J. Appl. Earth Observ. Geoinf., vol. 52, pp. 532–541, Oct. 2016.
[29] J. Reitberger, C. Schöntr, P. Krzystek, and U. Stilla, “3D segmentation of single trees exploiting full waveform LiDAR data,” ISPRS J. Photogramm. Remote Sens., vol. 94, no. 6, pp. 561–574, Nov. 2017.
[30] X. Lu, Q. Guo, W. Li, and J. Flanagan, “A bottom-up approach to segment individual deciduous trees using leaf-off lidar point cloud data,” ISPRS J. Photogramm. Remote Sens., vol. 94, pp. 1–12, Aug. 2014.
[31] I. Shendryk, M. Broich, M. G. Tulbure, and S. V. Alexandrov, “Bottom-up delineation of individual trees from full-waveform airborne laser scans in a structurally complex eucalypt forest,” Remote Sens. Environ., vol. 173, pp. 69–83, Feb. 2016.
[32] W. Chen, X. Hu, W. Chen, Y. Hong, and M. Yang, “Airborne LiDAR remote sensing for individual tree forest inventory using trunk detection-aided mean shift clustering techniques,” Remote Sens., vol. 10, no. 7, p. 1078, 2018.

[33] M. Ester, H. Kriegel, J. Sander, and X. Xu, “A density-based algorithm for discovering clusters in large spatial databases with noise,” KDD, vol. 96, pp. 226–231, Aug. 1996.

[34] S. Lamprecht, J. Stoffels, S. Dotzler, E. Haß, and T. Udelhoven, “ATrunk—An ALS-based trunk detection algorithm,” Remote Sens., vol. 7, no. 8, pp. 9975–9997, 2015.

[35] S. Tao, F. Wu, Q. Guo, Y. Wang, W. Li, B. Xue, X. Hu, P. Li, D. Tian, C. Li, H. Yao, Y. Li, G. Xu, and J. Fang, “Segmenting tree crowns from terrestrial and mobile LiDAR data by exploring ecological theories,” ISPRS J. Photogramm. Remote Sens., vol. 110, pp. 66–76, Dec. 2015.

[36] L. I. Duncanson, B. D. Cook, G. C. Hurtt, and R. O. Dubayah, “An efficient, multi-layered crown delineation algorithm for mapping individual tree structure across multiple ecosystems,” Remote Sens. Environ., vol. 154, pp. 378–386, Nov. 2014.

[37] K. Kennedy, K. Uzay, and C. Bayram, “Modeling stand parameters for Pinus brutia (Ten.) using airborne LiDAR data: A case study in Bergama,” J. Appl. Rem. Sens., vol. 14, no. 2, Oct. 2019, Art. no. 022205.

[38] Z. Hui, S. Jin, P. Cheng, Y. Y. Ziggah, L. Wang, Y. Wang, H. Hu, and Y. Hu, “An active learning method for DEM extraction from airborne LiDAR point clouds,” IEEE Access, vol. 7, pp. 89366–89378, Jul. 2019.

XIAOHU WANG received the B.S. degree in electronic information engineering from the Hunan University of Technology and Business, China, in 2003, and the M.S. degree in circuits and systems from Hunan University, China, in 2011. He is currently pursuing the Ph.D. degree in forestry engineering automation with Northeast Forestry University, China. His current research interest includes LiDAR point cloud processing.

YIZHUO ZHANG received the B.S. degree in computer science, the M.S. degree in control theory and control engineering, and the Ph.D. degree in mechanical design and theory from Northeast Forestry University, China, in 2001, 2003, and 2008, respectively. He was a Visiting Scholar with the Georgia Institute of Technology, USA, from 2009 to 2010. He is currently a Professor with Northeast Forestry University. He is also major in UAV remote sensing and global climate change.

ZE LUO received the B.S. degree in automation and the M.S. degree in computer application technology from the Central South University of Forestry and Technology, China, in 2012 and 2016, respectively. He is currently pursuing the Ph.D. degree in Forestry engineering automation with the Northeast Forestry University, Harbin, China. His current research interest includes LiDAR point cloud processing.

* * *