



Towards low vegetation identification: A new method for tree crown segmentation from LiDAR data based on a symmetrical structure detection algorithm (SSD)

Langning Huo^{*}, Eva Lindberg, Johan Holmgren

Department of Forest Resource Management, Swedish University of Agriculture Sciences, SE-901 83 Umeå, Sweden

ARTICLE INFO

Editor: Jing M. Chen

Keywords:

Individual-tree segmentation
Airborne laser scanning
Terrestrial laser scanning
Symmetrical structure
Crown base height
Low vegetation

ABSTRACT

Obtaining low vegetation data is important in order to quantify the structural characteristics of a forest. Dense three-dimensional (3D) laser scanning data can provide information on the vertical profile of a forest. However, most studies have focused on the dominant and subdominant layers of the forest, while few studies have tried to delineate the low vegetation. To address this issue, we propose a framework for individual tree crown (ITC) segmentation from laser data that focuses on both overstorey and understorey trees. The framework includes 1) a new algorithm (SSD) for 3D ITC segmentation of dominant trees, by detecting the symmetrical structure of the trees, and 2) removing points of dominant trees and mean shift clustering of the low vegetation. The framework was tested on a boreal forest in Sweden and the performance was compared 1) between plots with different stem density levels, vertical complexities, and tree species composition, and 2) using airborne laser scanning (ALS) data, terrestrial laser scanning (TLS) data, and merged ALS and TLS data (ALS + TLS data). The proposed framework achieved detection rates of 0.87 (ALS + TLS), 0.86 (TLS), and 0.76 (ALS) when validated with field-inventory data (of trees with a diameter at breast height ≥ 4 cm). When validating the estimated number of understorey trees by visual interpretation, the framework achieved 19%, 21%, and 39% root-mean-square error values with ALS + TLS, TLS, and ALS data, respectively. These results show that the SSD algorithm can successfully separate laser points of overstorey and understorey trees, ensuring the detection and segmentation of low vegetation in forest. The proposed framework can be used with both ALS and TLS data, and achieve ITC segmentation for forests with various structural attributes. The results also illustrate the potential of using ALS data to delineate low vegetation.

1. Introduction

During the past two decades, the potential of Light Detection and Ranging (LiDAR) in forest applications has been revealed by both scientific research and commercial products. One major application of LiDAR is forest data acquisition at a single-tree level, e.g. individual tree height, diameter at breast height (DBH), crown diameter, stem, and crown volume. Studies have indicated that reliable estimations of these forest parameters can be achieved using different scanning platforms, including airborne laser scanning (ALS), terrestrial laser scanning (TLS), mobile terrestrial laser scanning (MLS), and drone laser scanning (Liang et al., 2018; Wang et al., 2016; Kaartinen et al., 2012; Holmgren et al., 2019; Tao et al., 2015; Hyyppä et al., 2020; Li et al., 2019). To obtain estimates at a single-tree level, individual tree crown (ITC) segmentation

from the point cloud is a crucial step, for which numerous algorithms have been developed. Individual-tree detection and segmentation methods for ALS data include (1) raster-based methods, e.g. watershed segmentation (Tang et al., 2007; Ene et al., 2012), template matching (Huo and Lindberg, 2020), and deep learning (Windrim and Bryson, 2020; Chen et al., 2021; Zhu et al., 2021), and (2) point-based methods e.g. region growing (Solberg et al., 2006; Hyyppä et al., 2001), k-means clustering (Lindberg et al., 2013), and mean shift clustering (Ferraz et al., 2010; Melzer, 2007). These algorithms are able to detect dominant forest trees. Point-based methods generally require higher measurement densities than raster-based methods since the risk of occlusion is higher for objects below the top-most canopy. Hence, raster-based methods can be used for lower measurement densities than needed for point-based methods, but the latter are also capable of detecting small trees below

^{*} Corresponding author.

E-mail address: langning.huo@slu.se (L. Huo).

<https://doi.org/10.1016/j.rse.2021.112857>

Received 26 April 2021; Received in revised form 16 December 2021; Accepted 18 December 2021

Available online 3 January 2022

0034-4257/© 2021 The Authors. Published by Elsevier Inc. This is an open access article under the CC BY license (<http://creativecommons.org/licenses/by/4.0/>).

the top-most canopy.

With the development of scanners, more dense ALS point clouds have become available for more accurate ITC segmentation. Algorithms have been developed to detect subdominant trees in forest with more complex vertical layers, and various improvements have been made to increase the detection rate in the lower layers of the canopy. For example, the performance of the mean shift algorithm has been compared with different kernel shapes, adaptiveness, and weighting (Xiao et al., 2019), revealing that a crown-shaped kernel generates better segmentation, while weighting and adaptiveness have little influence on the performance. Ferraz et al. (Ferraz et al., 2012) have designed a statistical approach to calculate the suitable kernel bandwidth in the mean shift algorithm for ground vegetation, understory and overstory trees. Lindberg et al. (Lindberg et al., 2014) have improved the k-means clustering algorithm by using an ellipsoidal tree model. Williams et al. (Williams et al., 2020) have proposed an improved k-means algorithm, called the multiclass graph cut (MCGC) approach, by using local 3D geometry and density information. Ma et al. (Ma et al., 2020) have designed a region growing algorithm, morphology segmentation, and k-means algorithm as different steps for overall segmentation. Their region-growing algorithm is used first, for the dominant trees, then morphology is used to determine whether the segmentation needs to be refined, and finally the k-means algorithm is used for any necessary refinement. Dai et al. (Dai et al., 2018) have further refined under-segmentations in the spatial and multispectral domains of the mean shift segmentation algorithm. Commonly, these algorithms start from treetop detection, i.e., tree crowns are delineated from top to bottom.

Bottom-up designs for ITC segmentation from ALS point clouds have also been developed, based on dense point clouds that provide more points from the stems. As these algorithms do not depend on treetop detection, they perform well in detecting subdominant trees in canopies with more complex structures. For example, stems have been detected and reconstructed from a watershed segmentation combined with a RANSAC-based estimation of the stem points, and ITC segmentation then implemented using a normalized cut segmentation (Reitberger et al., 2009). Shendryk et al. (Shendryk et al., 2016) proposed a 3D random walk algorithm, with the segmentation starting from trunk detection. Using detected trunks as seed points, tree crowns are segmented into different trees according to the weights calculated by conditional Euclidean distance clustering.

Although algorithms have been developed that are reliable in more heterogeneous forests, attention has been focused on the dominant and subdominant trees; fewer algorithms have focused on low vegetation (Ferraz et al., 2012; Harikumar et al., 2019; Paris et al., 2016), especially the small trees below the top canopy. Detection of small trees below the top-most canopy improves estimations of diameter distributions and may also improve estimations of tree species and other tree attributes by defining the borders of the tree crowns better. Although these small trees do not contribute as much as dominant and subdominant trees to the volume or biomass, they are still an important component of ecological equations. For example, the variety of tree sizes can reveal the biodiversity of a forest. The complexity of the forest structure can also be used as an indicator for sustainable forest management. Compared with ALS data, TLS and MLS data have a higher capacity to capture low vegetation. However, few algorithms have been developed and validated specifically for the detection of small trees. Therefore, this study proposes an innovative ITC segmentation algorithm for accurately detecting low vegetation.

The main novelties of the proposed ITC segmentation framework are: (1) testing the symmetrical structure of individual trees to classify points as overstory/understory trees (the SSD algorithm); and (2) removing points belonging to overstory trees to simplify segmentation of the low vegetation. The performance of the proposed framework was tested on boreal forests with different stem densities and different complexities of vertical layers. In addition, the framework was validated with different datasets, including ALS, TLS, and merged point clouds from ALS and TLS

data (ALS + TLS). A visual interpretation was carried out to validate the low vegetation identified from the ALS + TLS data, which was then used as reference data for the TLS and ALS segmentations. The capacity of ALS to detect low vegetation was quantified by comparing it with the ALS + TLS segmentation.

2. Materials

2.1. Field data

The study area, Krycklan (64°14'N, 19°50'E), was located in the north of Sweden, within boreal forest (Fig. 1). The common tree species included Scots pine (*Pinus sylvestris*), Norway spruce (*Picea abies*), and birch (*Betula pendula* and *Betula pubescens*). A field inventory of 80 m square plots was conducted in 2016. Each 80 m × 80 m plot was divided into 16 subplots of 20 m, and the centers of the subplots were measured using a Differential Global Navigation Satellite System in the SWEREF99 coordinate system (EPSG: 3006). We used a combination of TLS and a trilateration system integrated with a digital caliper to estimate the position of each tree within each plot from the field inventory.

Within each subplot, all trees with a DBH ≥ 4 cm were inventoried for DBH and species. We took manual measurements of the stem diameters using a digital caliper integrated with an ultrasonic trilateration system (DP POSTEX, www.haglofsweden.com). The ultrasonic trilateration system measured the local coordinates for the tree positions with sufficient accuracy when limited to short ranges (approx. 10–15 m). Tree height was also measured for a random selection of sample trees.

The heights and DBH of the sampled trees were used to examine the allometric relationships between height and DBH, using Eq. (1) as suggested by Oinas and Sikannen (2000):

$$H = 1.3 + \exp\left(a - \frac{b}{DBH + 4} - \frac{c}{(DBH + 4)^2}\right) \quad (1)$$

where H is the tree height (m), DBH is tree diameter at breast height (cm), and a, b, and c are coefficients derived for pine, spruce, and deciduous trees, respectively. Ninety-one pine, 237 spruce, and 66 deciduous trees were used for the regression. The heights of all the trees were then calculated from the DBH using the allometric equation.

For accuracy analysis, the 20 m × 20 m subplots were used instead of the 80 m × 80 m plots, to take into account the variety of attributes found between the 16 subplots within one plot, e.g. species composition, and vertical structures. We did not use subplots that were selectively cut between the time when we obtained the ALS and TLS data. In total, 251 subplots from 23 plots were used in the study. Subsequently, “plot” refers to the 20 m × 20 m subplots. Plot data are presented as density level (Table 1), the number of canopy layers (Table 2), and tree species composition (Table 3). We classified the plots into low, medium, and high stem density, and had an approximately even number of plots in each category. The performance of ITC segmentation was compared across the different stem density classes.

We also summarized the condition of the vertical structure of the plots (Table 2). Assuming H_{max} is the largest tree height in a plot, we defined the first/s/third layers as containing trees with heights $\geq \frac{2}{3}H_{max}$, $\frac{2}{3}H_{max} - \frac{1}{3}H_{max}$, and $< \frac{1}{3}H_{max}$, respectively, as a simplified means of classifying trees into dominant/subdominant/understory categories. We thus categorized plots into L-1 to L-5, with L-1 representing plots that only contained first-layer trees, L-2 representing plots that contained first- and second- layer trees, and L-3 to L-5 representing plots with trees in every layer. To emphasize the performance of ITC segmentation on low vegetation, we used L-3, L-4, and L-5 to refer to plots with ≤15%, 15–30%, and > 30% trees in the third layer. Plots with different layer categories could then be compared.

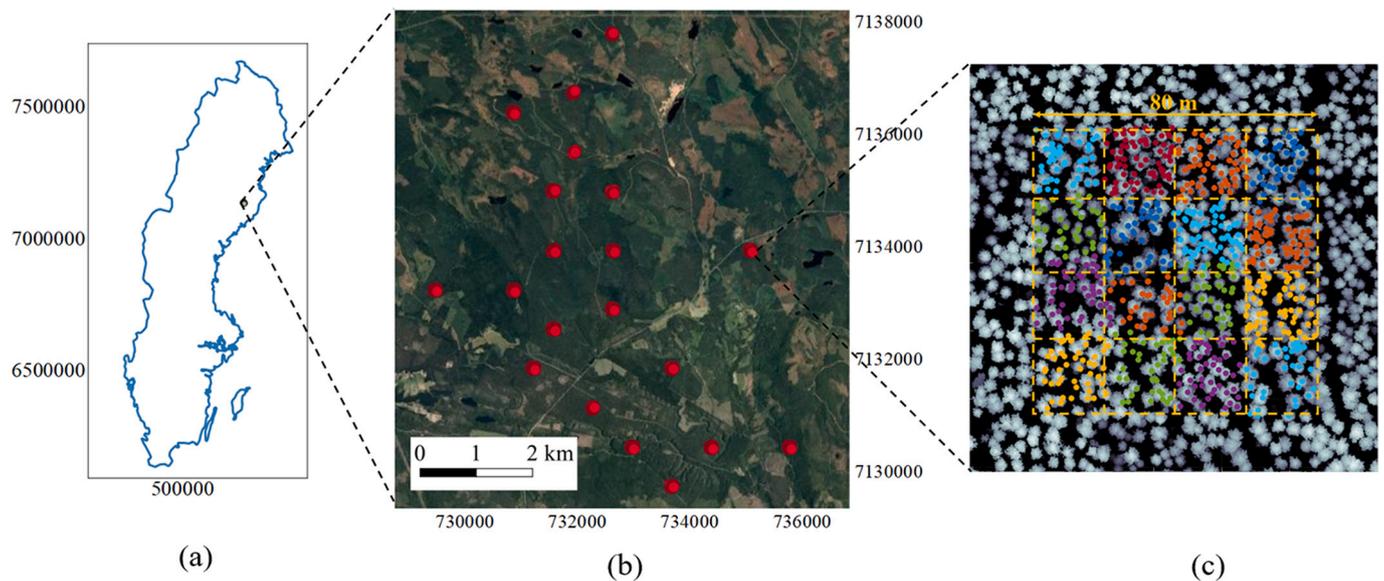


Fig. 1. The study area and field plots. (a) The study area in Sweden; (b) the plots in the field inventory; (c) the plots and subplots.

Table 1

The stem densities defined within the study plots.

Category	Density class	Threshold [stems/ha]	Number of plots	Stem density [stems/ha]		DBH [cm]		Tree height [m]	
				Mean	Std	Mean	Std	Mean	Std
D-1	Low	≤ 700	71	637	241	20.2	3.4	16.0	1.5
D-2	Medium	700–1100	28	704	306	23.4	5.4	17.5	3.2
D-3	High	> 1100	18	1383	732	14.7	3.1	12.0	1.9

Table 2

The forest layers defined within the study plots.

Category	Proportion of trees in each layer [%]			Number of plots	Stem density [stems/ha]		DBH [cm]		Tree height [m]	
	First layer	Second layer	Third layer		Mean	Std	Mean	Std	Mean	Std
L-1	> 0	0	0	15	427	142	24.3	2.3	18.1	0.9
L-2	> 0	> 0	0	44	567	196	22.3	4.8	17.2	2.4
L-3	> 0	> 0	$\leq 15\%$	77	940	343	17.5	2.7	14.2	1.6
L-4	> 0	> 0	15%–30%	71	1131	487	15.9	2.8	12.9	1.7
L-5	> 0	> 0	$> 30\%$	39	1316	666	13.0	2.4	10.9	1.5

Table 3

The species composition of the study plots.

Category	Species composition	Number of plots	Stem density [stems/ha]		DBH [cm]		Tree height [m]	
			Mean	Std	Mean	Std	Mean	Std
S-1	Pine	71	637	241	20.2	3.4	16.0	1.5
S-2	Spruce	28	704	306	23.4	5.4	17.5	3.2
S-3	Pine/spruce	57	1121	459	16.6	3.2	13.3	2.0
S-4	Pine/deciduous	37	901	376	15.3	2.5	12.8	1.6
S-5	Spruce/deciduous	38	1333	540	14.4	2.4	12.1	1.6
S-6	Pine/spruce/deciduous	18	1383	732	14.7	3.1	12.0	1.9

2.2. Laser data

On each 80 m \times 80 m plot, TLS data were collected using a Trimble TX8 from 4 \times 4 scanning stations, with each scanning station placed at the approximate center of each subplot (therefore with an internode distance of approximately 20 m). Individual point clouds from each scanning station were merged by co-registration using spheres, which were placed on adjacent scanning stations. Trees were detected and stem

diameters were estimated from the TLS data using a previously developed algorithm (Olofsson & Holmgren, 2016). We used the position of the scanning station, in the local coordinate system of the TLS data, as the starting position to search for the true position of the subplot center based on the manual measurements. We used an algorithm developed previously (Olofsson et al. 2008) to match the spatial patterns of the trees automatically with the manual measurements from the sub-plots and TLS data. The algorithm created one synthetic image of the tree

patterns for each data source, and matched the images to find the true position of each subplot.

Multispectral ALS data were acquired on June 28, 2019, with a RIEGL VQ-1560i-DW. This scanner records full-waveform although we only used the discrete returns in this study. This system uses two wavelengths: 532 nm (C1; green), with a beam divergence of 0.72 mrad, and 1064 nm (C2; NIR), with a beam divergence of 0.25 mrad. The flying altitude was 800 m above ground, the speed was 213 km/h, the overlap between the strips was 20%, the scan rate was 206 Hz with a field of view of 40°, and the pulse repetition rate was 1000 kHz per channel. This resulted in a scan width of 582 m and a measurement density of 24 pulses per m² and channel in each flight strip. The data was delivered in the SWEREF99 coordinate system (EPSG: 3006).

The point clouds from both ALS and TLS were normalized to remove the height of the ground. Points were classified as ground or non-ground using the Triangular Irregular Network (TIN) method implemented by the LAsTools software. The heights of the points were normalized using the heights from the Digital Elevation Model (DEM) generated. Points with heights lower than 0.5 m were removed. Fig. 2 exhibits some examples of the normalized Digital Surface Model (nDSM) of the plots with different categories.

The global coordinates (EPSG: 3006) were calculated for the points in the TLS data and then co-registered to ALS data. Each 80 m × 80 m plot contained 16 subplots, and the 16 subplots were scanned with the scanner at the center of each. The relative positions of the scanning centers were recorded, and their global coordinates were measured in the SWEREF99 coordinate system (EPSG: 3006). The relative and global coordinates of the 16 scanning centers were used to calculate the transformation matrix in the xy plane for each plot. The global coordinates of the laser points were then derived using the transformation matrices. Finally, the TLS point clouds were co-registered to ALS using the Iterative Closest Point (ICP) algorithm implemented in MATLAB.

A new ITC segmentation algorithm was designed for both ALS and TLS data, and the ALS and TLS point clouds merged. ITC segmentation was then compared between the different datasets.

3. Methods

3.1. The ITC segmentation framework

There were two main steps in the algorithm for individual tree

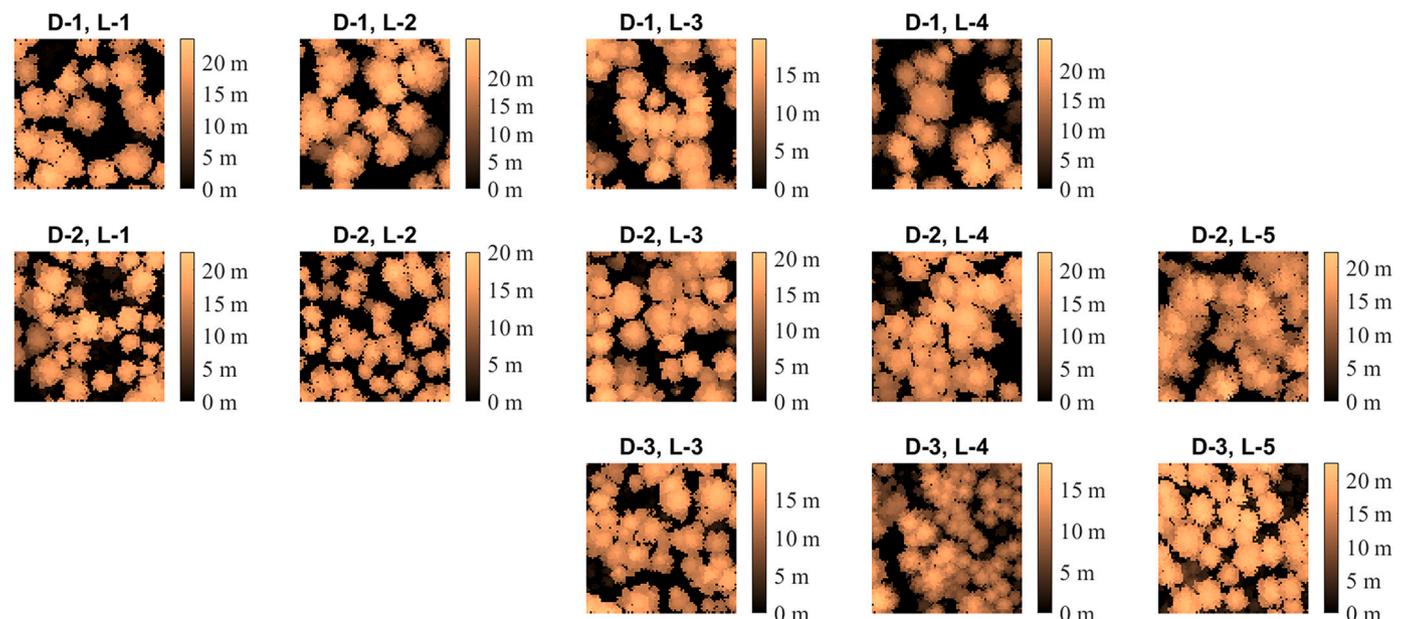


Fig. 2. The nDSM from the ALS data. Examples were selected to represent different density and layer categories.

segmentation (Fig. 3). First, the laser points for dominant trees (denoted as the P_{dominant} set) were detected and removed from the point cloud for the whole plot (denoted as the P_{all} set). P_{dominant} was identified from the surrounding point cloud by analyzing the symmetrical structure of the trees. The crown radius (CR) and crown base heights (CBH) were detected and used to create spaces for the dominant trees. Second, low vegetation was identified using Mean Shift Clustering of the point cloud without P_{dominant} . This step also involved criteria to remove clusters that belonged to the dominant trees.

The abbreviations and symbols used in the algorithm are listed in Appendix A, and the default parameter values are listed in Appendix B.

3.1.1. Detection and segmentation of dominant trees (SSD algorithm)

For this algorithm, the detection and segmentation of dominant trees was based on the symmetrical structure of individual trees, especially coniferous trees in boreal and temperate forests, with conical tree crowns and cylinder stems. In this study, P_{dominant} was identified by creating cylinder spaces for individual dominant trees, as in Fig. 4. The upper cylinder A was designed for tree crowns, with the CR identified as the radius of the cylinder. The lower cylinder B was designed for stems, with the CBH identified as the height of the cylinder. To obtain a proper space for P_{dominant} , the key parameters were CR and CBH.

The steps taken to detect the symmetrical structure of a tree and determine the CR and CBH values were as follows.

Step 1.1 Definition of target point clouds for individual trees

We detected local maxima from the nDSM smoothed by Gaussian filtering with a standard deviation (sigma) of 1 and denoted them as $T_{CHM} = \{t_k\}_{k=1}^K$, $t_k = (x_k, y_k, z_k)$. We set the seed points $S = \{s_n\}_{n=1}^N$ as a subset of T_{CHM} with $z_n \geq Th_{hs}$, i.e. a T_{CHM} is not considered to be a seed point if the height is lower than Th_{hs} . This is needed because small treetops may be detected as T_{CHM} when they stand in an open area, but their symmetrical structure may need different parameters compared with dominant trees. We set $Th_{hs} = \frac{1}{3} \text{Max}(\{z_k\}_{k=1}^K)$, to correspond to the upper 1/3 layer of the forest.

We denoted $P_{\text{target}} = \{p_i\}_{i=1}^I$ as the set of LiDAR points around the seed point s_n within a given radius Th_d , and with $z_i \leq z_n$, i.e. lower than the seed point s_n . Th_d should be at least as big as the largest possible crown radius, but as small as possible, to avoid under segmentation and speed up the calculation. We set Th_d as 3 m for our dataset.

Step 1.2 Equiangular sectorial voxelization

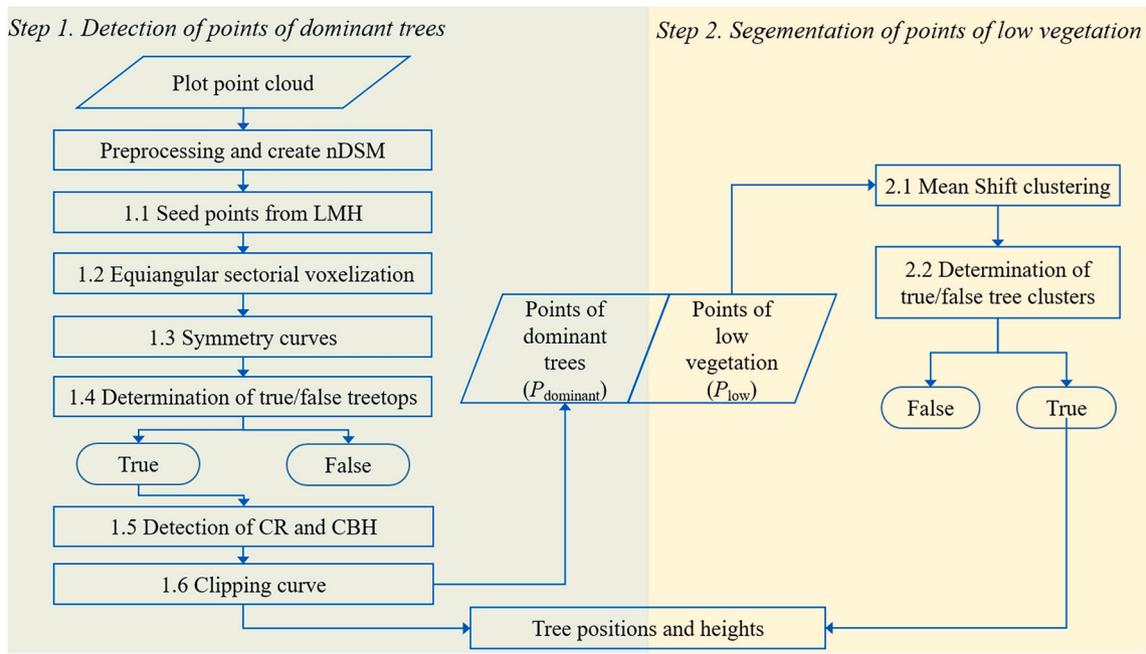


Fig. 3. The framework for the ITC segmentation algorithm.

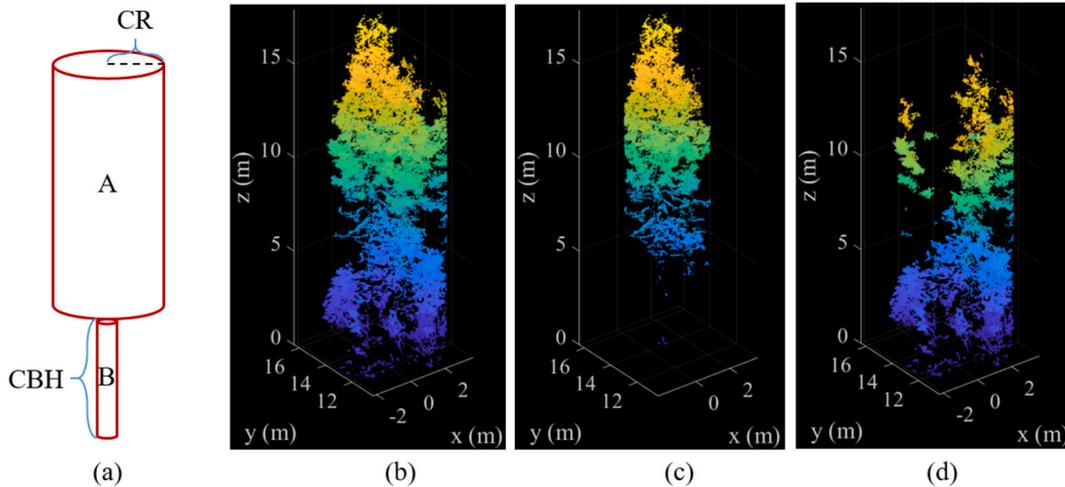


Fig. 4. An example of a cylinder space for ITC segmentation. (a) The cylinder space for an individual tree using CR and CBH as parameters. (b) All the points in a cylinder with a 3 m radius. (c) The points classified as P_{dominant} . (d) The rest of points.

The coordinates of the target point cloud were transformed from (x, y, z) to (z, d, α) using Eqs (1)–(2), and to (q, r, h) using Eqs (3)–(4):

$$\alpha = \tan^{-1} \frac{y}{x} \quad (1)$$

$$d = \sqrt{x^2 + y^2} \quad (2)$$

$$[h, r, q] = \text{Round}(M \times [z, d, \alpha, 1]^T) \quad (3)$$

$$M = \begin{bmatrix} \frac{1}{v_z} & 0 & 0 & 0 \\ 0 & \frac{1}{v_d} & 0 & 0 \\ 0 & 0 & \frac{1}{v_\alpha} & 0 \end{bmatrix} \quad (4)$$

where v_α , v_d and v_z are the voxel size. We set v_d to 0.5 m, v_z to 0.5 m, and

v_α to 30° . After voxelization, the space was divided into voxels as in Fig. 5. Voxels with points were denoted as occupied voxels, and those without points were denoted as empty voxels (Fig. 5).

Step 1.3 Plotting the symmetry curve

V_{ij} was the set of voxels with coordinates $h = H_i$ and $r = R_j$ (shown as the “crown ring” in Fig. 5d and e), with H_i being the height of the voxels and R_j the r coordinate of the voxels. We determined the “crown ring” V_{ij} to be symmetrical if more than 75% of voxels (Th_v) in set V_{ij} were occupied voxels. For ALS data, R_j was the largest R_j at H_i with a symmetrical V_{ij} . For TLS and ALS + TLS data, R_j was the largest R_j at H_i that made $V_{i1}, V_{i2}, \dots, V_{ij}$ all symmetrical. The curve was plotted for (R_j, H_i) , as in Fig. 6 (a-1, b-1). The symmetry curves looked similar to the side view of a tree, and usually revealed two types of tree shapes: a T-shaped tree (Fig. 6a), with a clear stem obviously visible in the symmetry curves, or an A-shaped tree (Fig. 6b), with a symmetrical crown down to the ground.

Step 1.4. Determination of true/false treetops

For each individual tree, a symmetrical structure should be detected

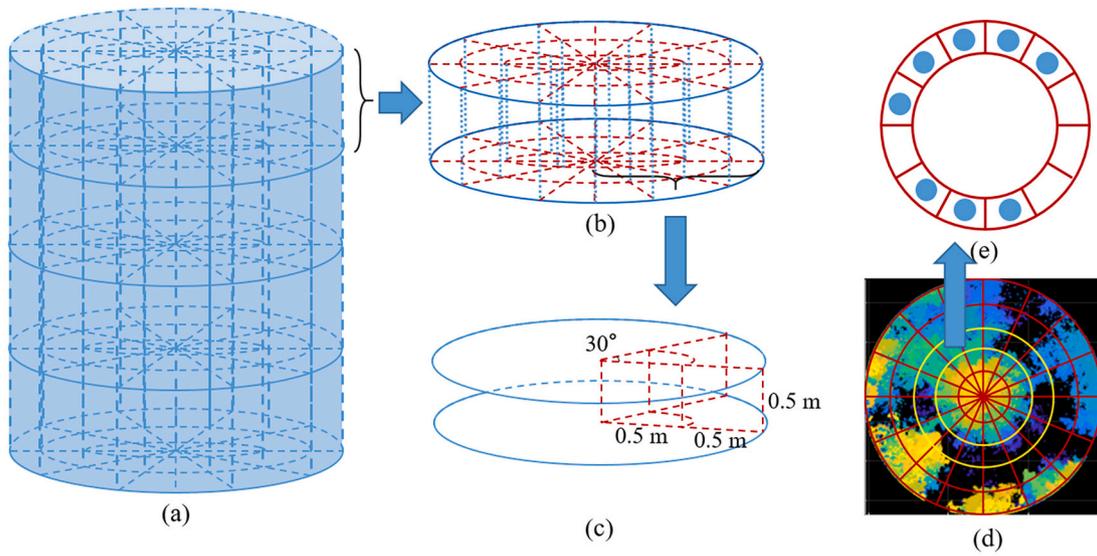


Fig. 5. The voxels generated after equiangular sectorial voxelization, and “crown rings” for testing the symmetrical structure. (a) A $4 \times 3 \times 12$ voxel space. (b) A $1 \times 3 \times 12$ voxel space, equal to one horizontal layer with height v_z . (c) A $1 \times 2 \times 1$ voxel space. (d) A “crown ring” (the space between the yellow lines) of a certain diameter and height. (e) A “crown ring” with 12 empty or occupied (indicated by blue dots) voxels. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

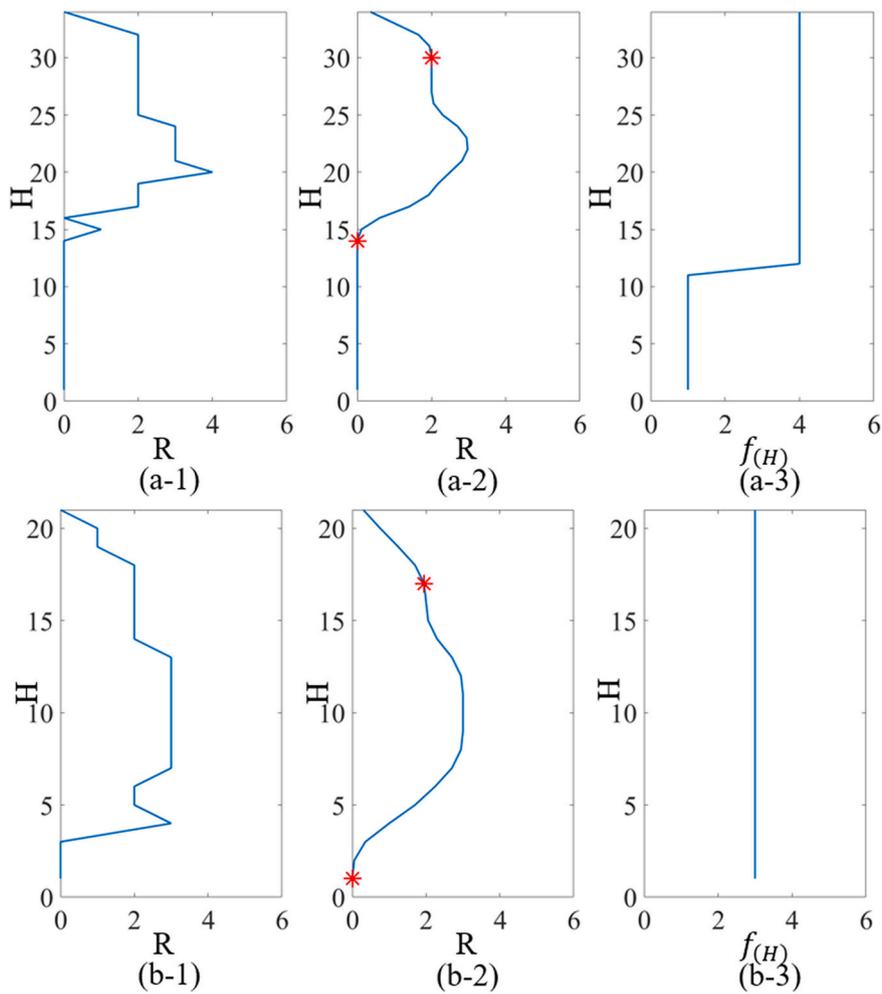


Fig. 6. Two examples of the symmetry curve (a-1, b-1), smoothed symmetry curve (a-2, b-2) and clipping curve (a-3, b-3), with (R_u, H_u) (upper red stars) and (R_l, H_l) (lower red stars). (a) An example of a pine with a funnel space. (b) An example of a spruce with a cylindrical space. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

from the position of the treetop as a horizontal symmetrical center. In step 1.1, we defined seed S from the local maxima of nDSM (T_{CHM}). However, seed S may contain local maxima caused by tall branches (denoted as false treetops, Fig. 7). Therefore, a subset $S_p = \{s_m\}_{m=1}^M$ was created for true treetops, and tree positions and heights were derived only from the true treetops, S_p . We determined a seed as S_p if (1) there was less than 50% zero R among the upper 1/3 tree height (for T-shaped trees); or (2) there was less than 50% zero R among the lower 1/2 tree height (for A-shaped trees).

Step 1.5 Detection of CR and CBH from symmetry curves

The symmetry curves were first smoothed, and from the curves, CR was determined as point (R_u, H_u) , and CBH as (R_l, H_l) (Fig. 6 a-2, b-2). This process was achieved using the following settings.

- Smoothing the symmetry curve. First, we replaced the local minimum and maximum values at different heights with the averaged values of the nearest neighbors, and then replaced zero values at different heights with the averaged values of nonzero neighbors. The curve was then smoothed using a Gaussian filter.
- Point (R_u, H_u) was taken to determine the radius of the upper crown. Detection began from the largest H values and continued towards the lower H values along the curve. The value of (R_u, H_u) was fixed at the first point when R no longer increased more than Th_s from H_i to H_{i+1} , and R_u represented the radius of the symmetrical crown. Th_s was set as 0.5 times the voxel size, i.e. 0.25 m in this study.
- Point (R_l, H_l) was at the lower part of the curve, indicating the lower edge of a tree crown with a height of H_l and a radius of R_l . Detection continues from H_u to a lower H , until the first local minimum R emerges, with the value (R_l, H_l) . If $R_l = 0$, the lower edge of the tree crown has been reached and there is only the stem under H_l . If $R_l \neq 0$, a crown might exist under H_l . To avoid the influence of a subtle local minimum R , the point (R_l, H_l) continues down the curve until R no longer increases more than Th_s (0.25 m) from H_i to H_{i+1} . If there is no local minimum for R , i.e. R keeps increasing or stays the same from H_c to 0, then we set $H_l = 0$, and R_l equals the largest R from H_c to 0.

Step 1.6 Creation of the clipping space

For each treetop, a clipping curve (Fig. 6 a-3, b-3) was created to include the space for the individual tree according to the (R_u, H_u) and (R_l, H_l) values. Laser points inside the clipped space were classified as P_{dominant} (Fig. 6). A voxel (h, r, q) was included in the clipped space if $r \leq f_{(h)}$. We determined a tree as an A-shaped tree when $H_l = 0$ or $R_u \leq R_l$, and otherwise as a T-shaped tree. Two types of clipping space were designed, as:

- ① A funnel space for a T-shaped tree, with the clipping curve as Eq. (5):

$$f_{(H_i)} = \begin{cases} \max(R_u, R_l) + \varepsilon, & H_i \geq H_l + \varepsilon \\ \max(R_l, \theta) + \varepsilon, & H_i < H_l + \varepsilon \end{cases} \quad (5)$$

where $\varepsilon = 2$ voxels (1 m) and $\theta = 1$ voxels (0.5 m) for our dataset. The value of ε was selected to include asymmetrical branches and the value of θ was selected to include points on the stems. The stems were usually asymmetrical with the local maxima of nDSM as the horizontal center, thus we set θ to include the points of stems even if they were not visible in the symmetry curves.

- ② A cylindrical space for an A-shaped tree, clipped by a straight line calculated as Eq. (6):

$$f_{(H_i)} = \max(R_u, R_l) + \varepsilon \quad (6)$$

When $H_l = 0$ or $R_u \leq R_l$, no clear stem was detected from the point cloud, e.g. a spruce with a tree crown down to the ground. Therefore, the space was not clipped for the stem but according to the crown diameters.

For each treetop, we classified points as P_{dominant} when they were inside the clipping space of the tree (Fig. 8). After point classification, those classified as P_{dominant} from all the treetops were removed from the point cloud simultaneously, and the rest of the points were denoted as P_{rest} . P_{rest} contained laser points that belonged to the low vegetation (denoted as P_{low}) and points from asymmetrical branches of dominant trees that were outside the clipping space (i.e. false negative points, P_{fn}).

3.1.2. Clustering of low vegetation

The main design of this step was to (1) cluster point clouds that belonged to the same trees using Mean Shift Clustering, and (2) remove clusters of P_{fn} .

Step 2.1 Mean Shift Clustering

The mean shift algorithm is a procedure for locating the maxima of a density function. It is a widely used algorithm for clustering point clouds in ITC segmentation using 3-D discrete data. We used a flat kernel and set the bandwidth to 1 (Th_{bw}). With this bandwidth, the algorithm performed well for grouping P_{low} and P_{fn} into different clusters, but could cause over-segmentation of P_{low} . Over segmentation of P_{low} usually occurred when the trees had several layers of branches within their crowns, appearing as several clusters in the vertical plane for a single tree. Therefore, we merged clusters that had centers close to each other in the (x, y) plane. To achieve this, we used the 2-D Mean Shift algorithm for the (x, y) values of the cluster centers. Two clusters were merged when their centers had a height difference smaller than Th_{hd} . We set Th_{hd} as 0.3 m.

Step 2.2 Determination of true/false tree clusters

The clusters were classified as low vegetation when the clusters met the following conditions.

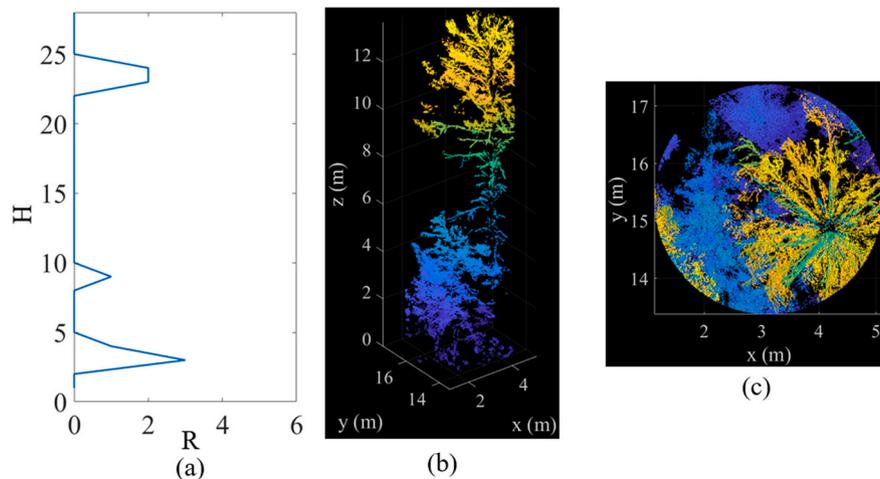


Fig. 7. An example of false treetops.

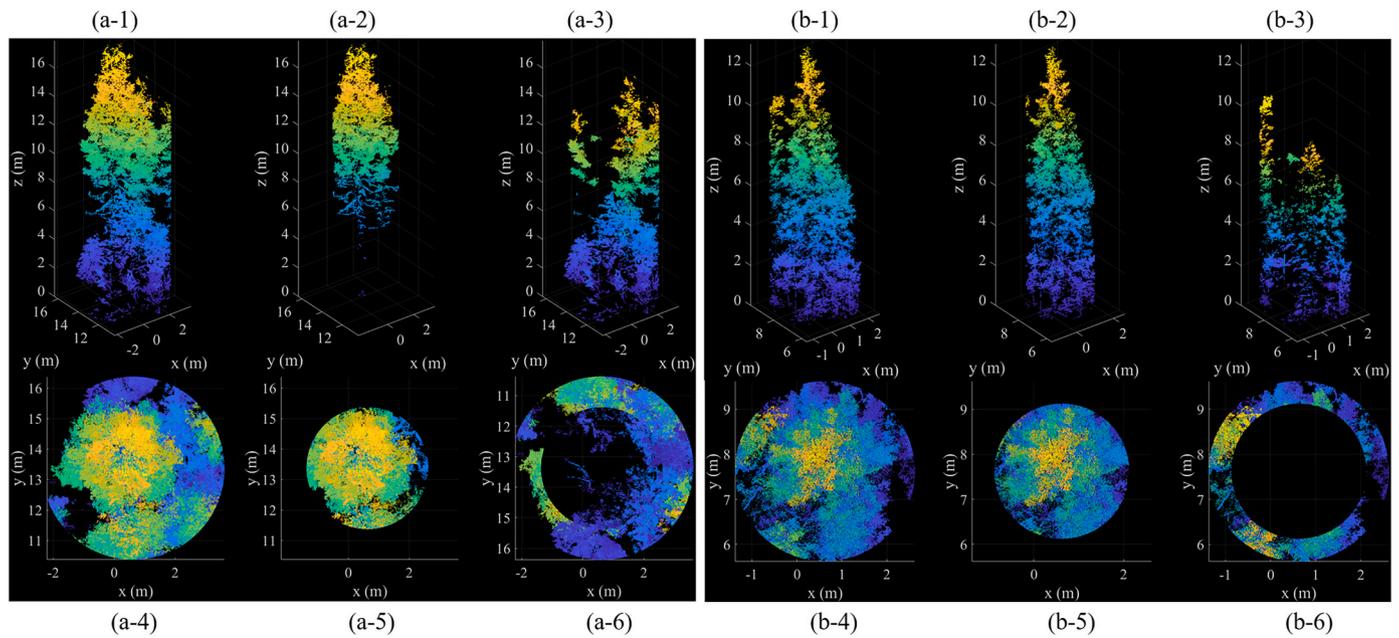


Fig. 8. Two examples of classifying P_{dominant} . (a) Point clouds of a pine. (b) Point clouds of a spruce. (1–3) Side view of the P_{target} , P_{dominant} and the rest of the points, respectively. (4–6) Top view of the P_{target} , P_{dominant} and the rest of the points, respectively.

- $z_{\text{min}} < Th_n \times z_{\text{max}}$, where z_{min} and z_{max} are the minimum and maximum z values of one cluster, respectively. This condition removes most of the P_{in} that correspond to branches of individual trees. We set $Th_z = \frac{2}{3}$.
- The area of one cluster projected to the horizontal plane is larger than Th_a . We calculated this area by computing the two-dimensional convex hull of all the points in one cluster. We set Th_a as 1 m^2 . This condition removed most of the P_{in} that belonged to stems.
- The number of points divided by the three-dimensional volume is larger than Th_n . We used 0.5 m voxels to represent the volume. We set Th_n to 20 for ALS + TLS and TLS data, and 5 for ALS data.

We also removed clusters with maximum heights lower than 1 m , because they were difficult to validate by visual interpretation. We only validated the accuracy for vegetation higher than 1 m . Points in the clusters were classified as P_{low} . Tree positions (cluster center) and heights (maximum heights of the points) were derived from the clusters

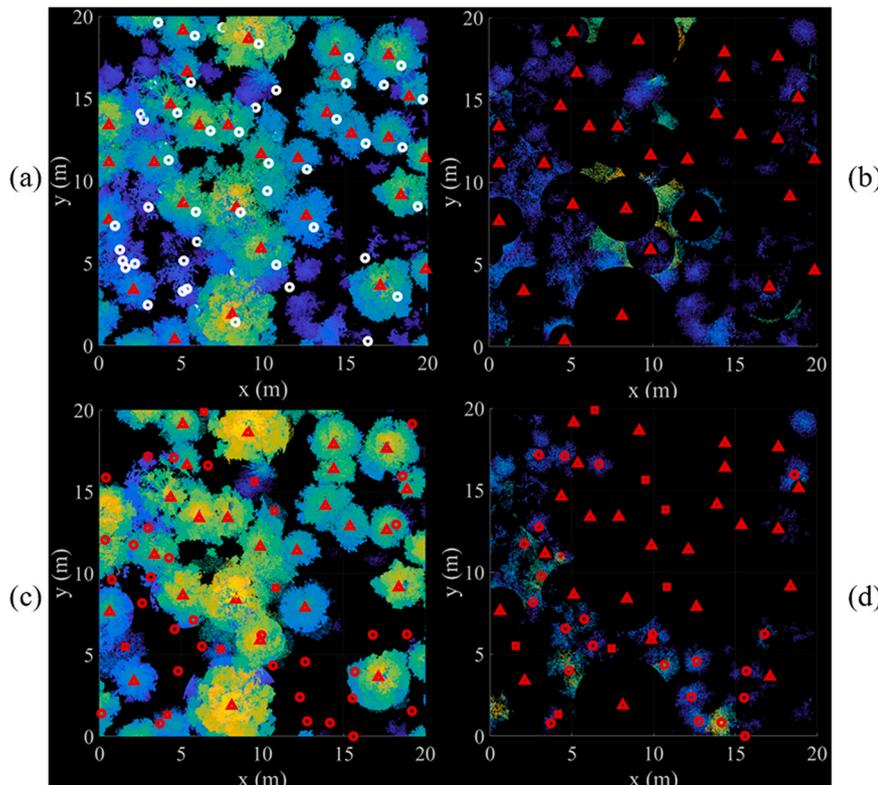


Fig. 9. Point clouds for an example plot. (a) The ALS + TLS data from a $20 \times 20 \text{ m}$ plot with a 5 m buffer. (b) P_{rest} obtained after Step 1.6. (c) The final P_{dominant} set after Step 2.2. (d) The final P_{low} in the third layer. White circles represent tree positions as measured by the field inventory. Red triangles represent true treetops, S_p . Red squares and circles represent the second- and third-layer tree positions detected by the algorithms, respectively. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

of P_{low} . Trees detected from 3.1.1 and 3.1.2 were trees detected in the entire plots (Fig. 9).

To improve the visual interpretation during the validation, we added another step to reclassify points of subdominant trees from P_{low} to $P_{dominant}$ according to height. Following the steps in 3.1.1, the treetops of subdominant trees were not detected as local maxima, thus they were classified as P_{low} . We reclassified the points as $P_{dominant}$ if the corresponding clusters had points in the second layer that corresponded to subdominant trees. After this reclassification, errors in understory-tree detection were easier to detect by visual interpretation of the third-layer trees, with third-layer trees in $P_{dominant}$ as omission errors and over-segmentation in P_{low} as commission errors.

3.2. Validation

The field inventory only measured trees with a DBH ≥ 4 cm, while the new algorithm could detect smaller trees. We used the different methods to compare the performance between the first two layers and the third layer. Trees in the first two layers were validated by matching with the field inventory data, while the third-layer trees were validated by visual interpretation of the point cloud.

3.2.1. Matching with reference data

To validate the detection accuracy, tree positions derived from laser data (laser-trees) were matched to the field measurements (field-trees). The laser-trees were first linked to all possible field-trees within a 3 m distance and with a height difference smaller than 5 m. All the possible linked pairs were then sorted by distance. Pairs with smaller distances were prioritized for retention, unless a tree was already linked to another tree by an even smaller distance. The numbers of true positives (TP), false negatives (FN), and false positives (FP) were counted. TP represented the matched laser-trees/field-trees, FN the unmatched field-trees (omission error), and FP the unmatched laser-trees (commission error). The detection rate (DR_A) for all trees was calculated to show how many field-trees could be detected by the algorithm, using Eq. (7). We also calculated the DR of the first (DR_1), second (DR_2), and third (DR_3) layers, respectively, to highlight any differences in performance between the layers. The recall (R), precision (P), and F-scores (F) were calculated for the first- and second-layer trees using Eqs. (8)–(10). Unmatched laser-trees in the third layer could relate to unmeasured small trees in the reference data, so the third-layer trees had to be excluded to calculate P .

$$DR_{A,1,2,3} = \frac{TP}{TP + FN} \quad (7)$$

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

$$P = \frac{TP}{TP + FP} \quad (9)$$

$$F = 2 \times \frac{R \times P}{R + P} \quad (10)$$

3.2.2. Visual interpretation of the low vegetation

The validation of third-layer trees was conducted by visual interpretation of the point cloud. We counted the number of clusters, as:

N_1 = Number of undetected trees in $P_{dominant}$, caused by an error in the symmetry curve (Step 1.5). N_1 corresponded to the creation of too much space for the dominant trees, therefore including under-growth. In this case, the estimated CR was larger or the estimated CBH lower than the true value, which misclassified P_{low} as $P_{dominant}$. Fig. 10 C1 illustrates an example of this.

N_2 = Number of undetected third-layer trees in $P_{dominant}$, caused by errors in clustering (Step 2.2).

N_3 = Number of false clusters in P_{low} , caused by errors in the symmetry curve (Step 1.5). N_3 corresponded to the creation of too little space for the dominant trees, excluding the branches. In this case, the estimated CR was smaller or the estimated CBH higher than the true value, which misclassified $P_{dominant}$ into P_{low} . Some of these clusters could be reclassified as $P_{dominant}$ in step 2.2, while the rest became false clusters in P_{low} . Fig. 10 C3 illustrates an example of this.

N_4 = Number of over-segmented clusters, caused by errors in clustering (Step 2.1).

N_5 = Number of under-segmented clusters, caused by errors in clustering (Step 2.1, Fig. 10 C2).

N_e = Number of laser-trees in the third layer.

Because of the limitations of visual interpretation, we did not count clusters with an area obviously less than $1 \times 1 \times 1 \text{ m}^3$. This was consistent with the condition set in Step 2.2. The algorithm had the capacity to detect trees smaller than $1 \times 1 \times 1 \text{ m}^3$, but the visual interpretation of such small trees was difficult and could lead to a biased validation; thus they were excluded. From the counted numbers, we calculated the omission error (OE), commission error (CE), true number of third-layer trees (N_a), bias in number of detected trees (Bias), and root-mean-square error (RMSE), using Eqs. (11)–(15):

$$OE = N_1 + N_2 + N_5 \quad (11)$$

$$CE = N_3 + N_4 \quad (12)$$

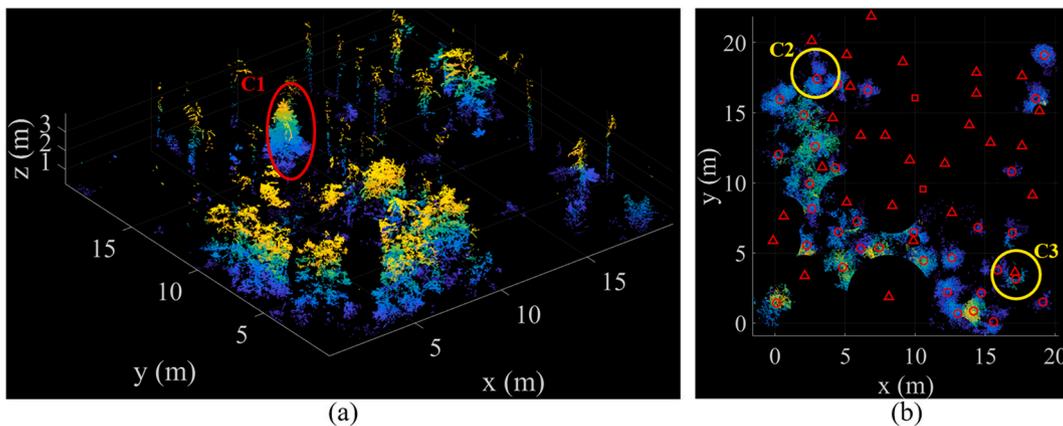


Fig. 10. Examples of possible errors in the algorithm. (a). $P_{dominant}$ with $z < 4$ m. (b). P_{low} . Red triangles, squares, and circles represent positive seeds, and the second- and third-layer tree positions detected by the algorithm, respectively. C1 is an example of undetected third-layer trees in the $P_{dominant}$. C2 is an example of under-segmentation of the clusters. C3 is an example of clusters belonging to branches of dominant trees. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

$$N_a = N_c + OE - CE \quad (13)$$

$$\text{Bias} = N_c - N_a \quad (14)$$

$$\text{RMSE} = \sqrt{\frac{\sum_{k=1}^n (N_c - N_a)^2}{n}} \quad (15)$$

We also represented the performance of the new SSD algorithm using the classification accuracy between P_{dominant} and P_{low} at the individual-tree level (CA), using Eq. (16):

$$\text{CA} = 1 - \frac{N_1 + N_3}{\text{Number of positive seeds}} \quad (16)$$

Because of the workload of visual interpretation, we only validated 121 plots selected by systematic sampling. Plots with the same species composition category (S-1 to S-6), density category (D-1 to D-3), and layer category (L-1 to L-5) were regarded as the same group. At most, three plots were randomly selected from each group if there were more than three plots in the group. We conducted a visual validation of the 121 plots for the ALS + TLS data, and calculated OE, CE, CA, Bias and RMSE. For the TLS and ALS data, only the number of third-layer trees was validated, by a comparison of N_a , and then Bias and RMSE were calculated.

3.2.3. Method comparison and sensitivity analysis

We compared the results from the SSD algorithm with those from a commonly used algorithm (Wang et al., 2016; Kaartinen et al., 2012), i. e. local maxima detected from the nDSM. The latter is a simple and commonly used method for finding tree locations (Matsugami, 2012; Khorrami et al., 2018; Ma et al., 2021), and is also widely used as the first in a series of algorithms to improve segmentation, e.g. followed by k-means clustering (Morsdorf et al., 2003), watershed segmentation (Zhang et al., 2014), template matching (Huo and Lindberg, 2020), and Silva2016 (Silva et al., 2016) implemented in the lidR package. As with the SSD algorithm, initially local maxima of the nDSM were detected, then, in subsequent steps, false treetops were removed and understory trees detected by testing symmetry structures. Comparing the SSD algorithm and local maxima detection highlighted potential improvements for later steps in the SSD algorithm. The detected trees were matched with the field reference data and the $DR_{A, 1, 2, 3}$ calculated.

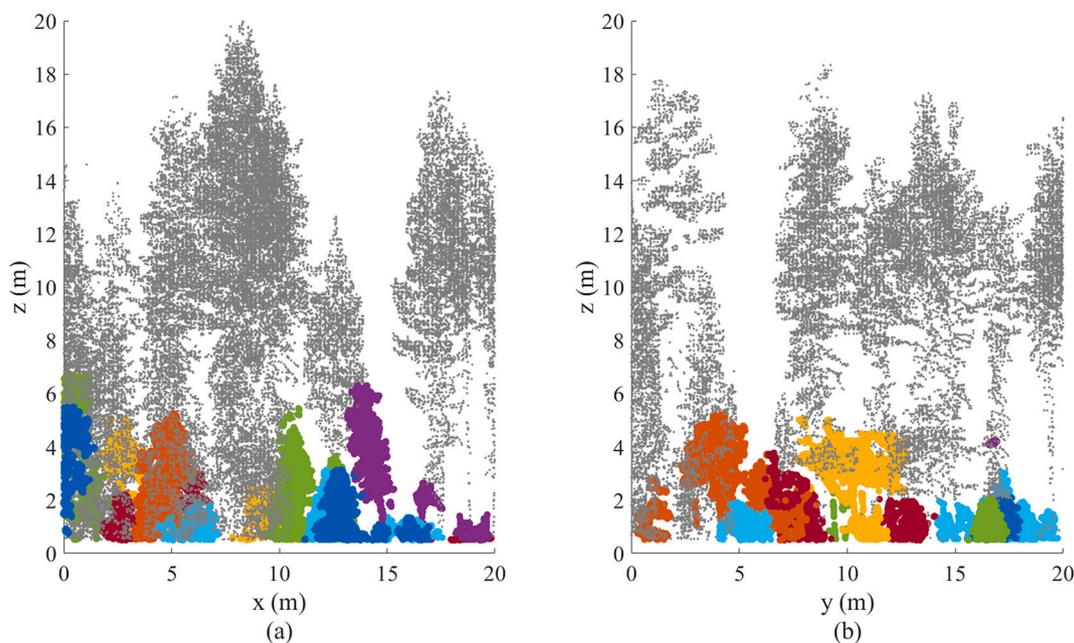


Fig. 11. Examples of low vegetation segmentation (a – b). Gray dots represent the point clouds of dominant trees, and the colored dots clusters of low vegetation.

We then conducted a sensitivity analysis to see whether changing the parameter settings altered the accuracy of the results, and tested whether the performance depended on using optimized parameters. The parameter values were changed one at a time, and the detection rates determined by matching the field reference data and the numbers of detected third-layer trees. The sensitivity analysis was not used for parameter tuning, so the results from 3.2.1 and 3.2.2 were not updated to the best possible results derived from the sensitivity analysis.

4. Results

Figs. 11 and 12 illustrate examples of the segmentation results.

4.1. ITC segmentation of trees with $DBH \geq 4$ cm

When averaging all the plots, the framework achieved a value of 0.87 for DR_A when using ALS + TLS data, 0.86 DR_A when using TLS data, and 0.76 DR_A when using ALS data. Similar to other ITC segmentation algorithms, the detection rates decreased when plots had higher densities and more vertical layers (Table 4). The performance of TLS data was the same or slightly lower than ALS + TLS data for the different categories. DR for the ALS data was obviously lower than the other two datasets, while P showed the opposite trend. This implied that ALS detection yielded higher omission errors but lower commission errors than ALS + TLS and TLS.

The detection rates were compared between plots with different complexities in the vertical layers (Fig. 13). The SSD algorithm exhibited the lowest DR in the second layer, and the highest in the third layers, especially in ALS data. The detection of third-layer trees was similar among L-3, L-4, and L-5. This implied that the proportion of trees in the third layer did not influence detection rates significantly. Compared with ALS + TLS and TLS data, ALS data showed obvious lower DR values in the second layer, but only slightly smaller DR values in the third-layer trees. Compared with the local maximum detection method (Fig. 13), the SSD algorithm was superior in detecting second- and third-layer trees.

4.2. ITC segmentation of third-layer trees

Visual interpretation was carried out for 121 sampled plots using

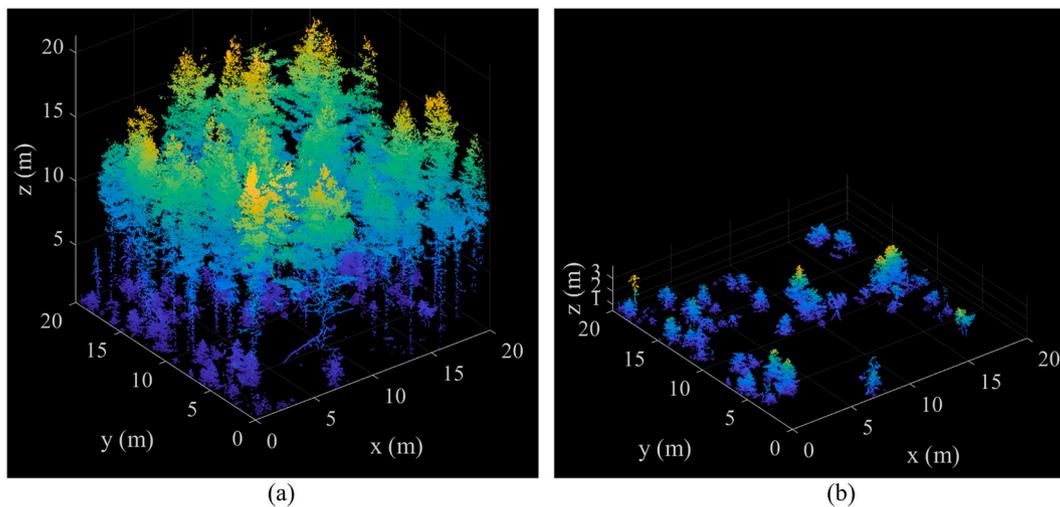


Fig. 12. Examples of $P_{\text{dominant}}/P_{\text{low}}$ separation. (a) A point cloud of the whole plot. (b) A point cloud of low vegetation.

Table 4

Average DR_A , R , P and F values for different categories of data.

Attribute	Category*	ALS + TLS				TLS				ALS			
		DR_A	R	P	F	DR_A	R	P	F	DR_A	R	P	F
Density	D-1	0.93	0.93	0.83	0.88	0.93	0.93	0.83	0.88	0.86	0.85	0.92	0.88
	D-2	0.87	0.88	0.89	0.89	0.86	0.87	0.88	0.87	0.75	0.75	0.94	0.84
	D-3	0.78	0.80	0.93	0.86	0.78	0.80	0.93	0.86	0.63	0.63	0.94	0.76
Layers	L-1	0.97	0.97	0.91	0.94	0.97	0.97	0.89	0.93	0.82	0.82	0.98	0.89
	L-2	0.94	0.94	0.87	0.90	0.93	0.93	0.85	0.89	0.82	0.82	0.93	0.87
	L-3	0.86	0.86	0.91	0.88	0.85	0.86	0.90	0.88	0.72	0.72	0.96	0.82
	L-4	0.84	0.85	0.89	0.87	0.84	0.85	0.89	0.87	0.73	0.73	0.94	0.82
	L-5	0.78	0.80	0.82	0.81	0.80	0.82	0.82	0.82	0.74	0.72	0.86	0.79
Species	S-1	0.95	0.96	0.92	0.94	0.94	0.94	0.92	0.93	0.81	0.81	0.96	0.88
	S-2	0.87	0.87	0.77	0.81	0.86	0.86	0.75	0.80	0.79	0.80	0.86	0.83
	S-3	0.83	0.84	0.89	0.86	0.84	0.84	0.88	0.86	0.69	0.69	0.92	0.79
	S-4	0.85	0.85	0.82	0.84	0.85	0.85	0.83	0.84	0.78	0.76	0.90	0.82
	S-5	0.79	0.82	0.93	0.87	0.79	0.81	0.92	0.86	0.70	0.71	0.96	0.82
	S-6	0.81	0.82	0.90	0.86	0.82	0.84	0.90	0.87	0.69	0.69	0.96	0.80
All plots		0.86	0.87	0.83	0.84	0.86	0.87	0.82	0.84	0.75	0.75	0.89	0.80

* Definitions of the different categories are given in Tables 1–3.

ALS + TLS data. The average RMSE was 3.0 trees (19% relative RMSE). The Bias of each plot had a distribution of -5 to 10 trees, with an average of 0.6 ± 2.3 trees ($3\% \pm 14\%$). The Bias was independent of the actual number of third-layer trees in the plots (Fig. 14). Plots with different stem densities and vertical structures had similar Bias, while spruce plots (S-2) yielded obvious over-estimations compared with the other plots (Table 5). The relative RMSE for TLS data was 21%, which was slightly higher than for ALS + TLS data. Using ALS data resulted in a -14% Bias and 39% RMSE, indicating an obvious under-estimation of the third-layer trees; the under-estimation was more severe with larger stem densities.

From the 121 plots sampled for third-layer validation, we obtained 4843 positive seeds from the ALS + TLS data for classifying P_{dominant} and P_{low} ; 232 of these had errors that led to the over- or under-estimation of the third-layer trees, i.e. a classification accuracy (CA) of 95%. CA was independent of stem density, slightly influenced by the complexity of the vertical structure, and significantly influenced by species composition (Table 5). The pure pine plots (S-1) had the highest CA values (98%), while the pure spruce plots (S-2) had the lowest CA values (91%).

The detection using TLS data performed similarly to ALS + TLS data in general (Table 6). TLS data yielded slightly higher Bias and RMSE than ALS + TLS data, and was also influenced by the species composition but not density or layers. ALS data illustrated obvious under-estimations of the third-layer trees (Table 6).

The sensitivity analysis involved changing the parameter values one at a time (Appendix C). Even with relatively large value discrepancies, only two parameters changed the performance significantly: (1) a voxel size smaller than 0.5 m, 0.5 m, and 30° for the TLS and ALS + TLS data (reasons for which are suggested in the discussion), and (2) a θ value of 0 in Eq. (5). θ in Eq. (5) was used to classify the points of stems as dominant trees even though they might not be symmetrical, and thus θ could not be set to 0 . We used $\theta = 1$ voxels (0.5 m) for the study, and using $\theta = 2$ voxels (1 m) in the sensitivity analysis produced similar results. All other parameter values resulted in detection rates changing from -0.11 to 0.07 compared with the default settings. This suggested that using optimal parameters was not crucial for the SSD algorithm's performance.

5. Discussion

We have presented an ITC segmentation framework for accurate understory-tree identification. The key technique is the SSD algorithm for segmenting point clouds of dominant trees and low vegetation. The space was divided into sectorial voxels and a symmetry curve was drawn from rings with different radii ("crown rings" in Fig. 5d and e). Points inside the voxels in the symmetry curve were defined and removed from the point cloud. This step not only reduced the computation needed for later clustering, but was also better at distinguishing lower branches of

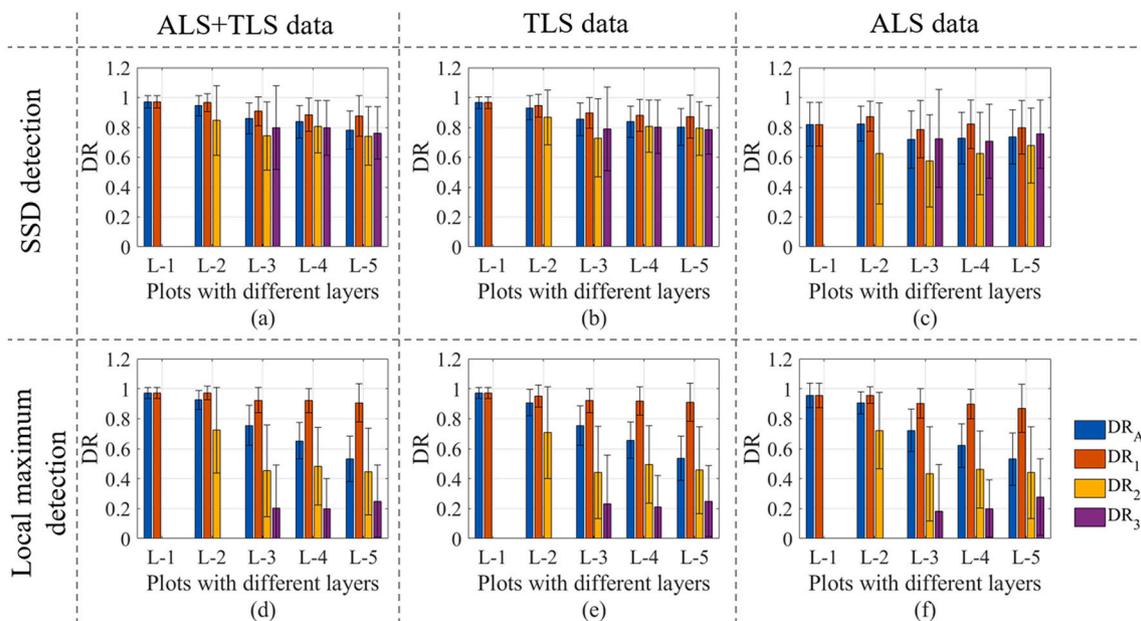


Fig. 13. Detection rates of plots with different numbers of layers by SSD detection (a, b, c) and local maximum detection (d, e, f). (a, d) Detection using ALS + TLS data. (b, e) Detection using TLS data. (c, f) Detection using ALS data.

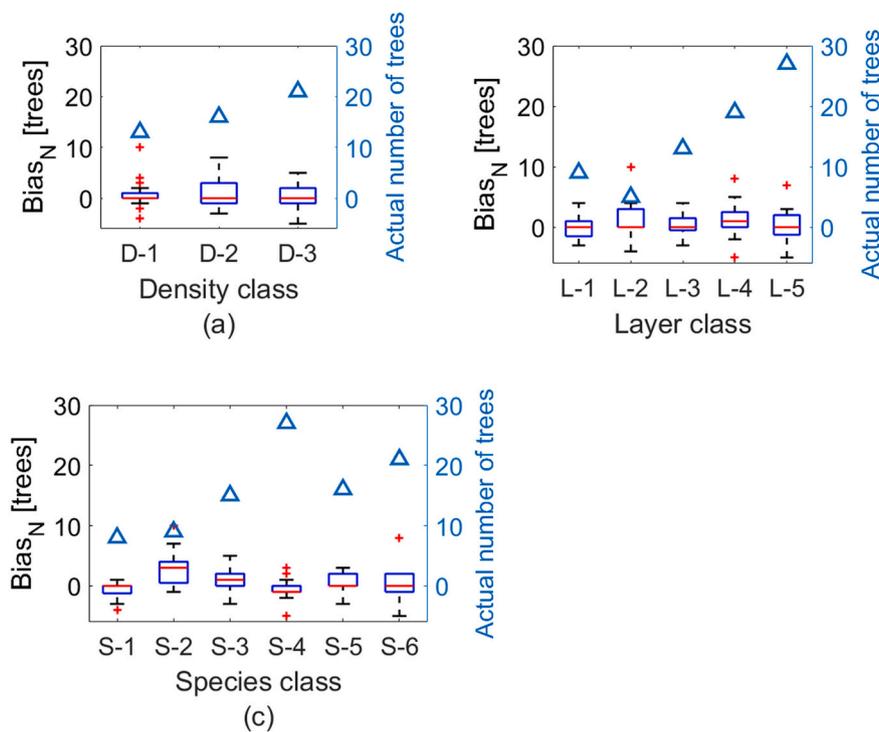


Fig. 14. Estimation bias and actual number of trees in plots from different categories. (a) Density classes. (b) Layer classes. (c) Species classes.

dominant trees and smaller trees, compared with clustering points for whole plots.

In the process of detecting symmetrical structures, an equiangular sectorial voxelization was carried out to generate voxels 0.5 m in height, 0.5 m in width, and with a 30° intersection angle between the voxel borders. This voxelization also reduced the computation time significantly. The scale of symmetry was important for the algorithm. Theoretically, a smaller voxel size could improve the accuracy of CR and CBH, but, in practice, it can also make the symmetry curve more sensitive to the asymmetrical branches inside a tree crown. For example,

0.5 m voxels can be used to describe the crown, which is symmetrical, but 0.1 m voxels would describe the individual branches, which may not be symmetrical. The sensitivity analysis (Appendix C) also showed that a smaller voxel size required more symmetry for a tree crown to be detected, resulting in more points being determined as unsymmetrical and classified as low vegetation.

Similar to other ITC segmentation algorithms (Wang et al., 2016), the detection accuracies of the SSD algorithm decreased with increasing stem density and more vertical layers. Species composition also influenced the performance, and Bias was larger for plots with spruce rather

Table 5
ITC segmentation results for third-layer trees from ALS + TLS data.

Attribute	Category	N _a [trees/ha]	CA [%]	OE		CE		Bias		RMSE	
				[trees/ha]	[%]	[trees/ha]	[%]	[trees/ha]	[%]	[trees/ha]	[%]
Density	D-1	322	95	14	4	33	4	16	5	57	18
	D-2	400	94	24	6	44	6	20	5	88	22
	D-3	528	95	34	6	36	6	11	2	80	15
Layers	L-1	225	97	25	11	25	11	0	0	57	25
	L-2	113	95	13	12	42	12	33	29	83	73
	L-3	325	96	20	6	29	6	11	3	54	17
	L-4	485	94	24	5	48	5	22	4	96	20
	L-5	665	94	35	5	36	5	7	1	69	10
Species	S-1	206	98	23	11	6	11	-18	-9	39	19
	S-2	223	91	11	5	84	5	88	39	109	49
	S-3	375	95	14	4	43	4	32	9	75	20
	S-4	683	96	30	4	20	4	-16	-2	89	13
	S-5	407	93	28	7	43	7	21	5	60	15
	S-6	513	95	39	8	45	8	7	1	60	12
All plots		391	95	24	6	38	10	17	4	73	19

Table 6
ITC segmentation results for third-layer trees from TLS and ALS data.

Attribute	Category	N _a [trees/ha]	TLS				ALS			
			Bias		RMSE		Bias		RMSE	
			[trees/ha]	[%]	[trees/ha]	[%]	[trees/ha]	[%]	[trees/ha]	[%]
Density	D-1	322	30	9	72	22	17	5	128	40
	D-2	400	9	2	99	25	-86	-21	146	36
	D-3	528	24	4	82	16	-120	-23	219	42
Layers	L-1	225	25	11	60	27	-25	-11	64	29
	L-2	113	27	24	71	63	-3	-3	83	73
	L-3	325	19	6	67	21	-72	-22	152	47
	L-4	485	15	3	101	21	-96	-20	181	37
	L-5	665	30	5	96	14	-30	-5	216	32
Species	S-1	206	-6	-3	43	21	-48	-23	82	40
	S-2	223	83	37	119	53	-61	-27	122	55
	S-3	375	26	7	74	20	-78	-21	181	48
	S-4	683	-1	0	108	16	-64	-9	207	30
	S-5	407	25	6	69	17	-33	-8	166	41
	S-6	513	20	4	72	14	-73	-14	197	38
All plots		391	23	6	81	21	-55	-14	153	39

than pine and deciduous trees. Compared with pine and birch (T-shaped trees), spruce trees have more asymmetrical branches closer to the ground (A-shaped trees), which is more challenging to distinguish from understory trees. We therefore applied two types of clipping curve based on the symmetrical radius, generating a funnel (Fig. 6a-3) or cylindrical (Fig. 6b-3) space for the dominant trees. A funnel space was more common for trees with a clear crown base above the ground or low vegetation, while a cylindrical space was found almost exclusively for spruces with branches close to the ground. For the funnel space, CBH accuracy was the key factor that influenced the detection of understory trees, while CR accuracy influenced the detection of subdominant trees, but not the understory trees. Only when a cylindrical space was generated did the accuracy of CR influence the detection of understory trees, because no understory trees could be detected within the CR. An underestimated CR would also exclude branches from dominant trees. When these excluded branches were close to the ground, they were detected as understory trees. For this reason, a bigger positive Bias was found for third-layer trees in the spruce stands than plots with other tree species.

Comparing results from the SSD algorithm and local maxima detection (Fig. 13), the SSD algorithm improved the detection in two ways. (1) False treetops were excluded from the seeds by testing the symmetry curve, which lowered the CE. (2) Subdominant trees, especially trees standing close to dominant trees, whose treetops were not detected as local maxima from the nDSM, could be separated from the dominant trees because their crowns were not part of the symmetrical crowns of the dominant trees. They could then be detected in Step 2, which

improved the DR of the subdominant trees. Nevertheless, the detection of second-layer trees still yielded lower DR values compared with other layers. Other algorithms could be used to improve this, for example, segmenting P_{dominant} values for individual trees again using k-means clustering.

We tested the framework on ALS + TLS, TLS and ALS data, and the results illustrated the relative robustness of the different scanning datasets. Compared with ALS + TLS data, TLS data exhibited almost the same or slightly lower accuracy of ITC segmentation. For the TLS data, many high branches were detected as local maxima because of the low point density of the treetops. Merging ALS data with the TLS data made the treetops more distinct in the point cloud, which decreased the error in Step 1.1. However, as mentioned above, testing the symmetry curve (Step 1.4) compensated for false treetops, which decreased the difference between ALS + TLS and TLS data. This also illustrates that the SSD algorithm was sufficiently robust to tolerate seeds that were not centered exactly in the crowns. ALS showed a lower capacity to detect second-layer trees than TLS data, but the detection of large third-layer trees (DBH \geq 4 cm) was similar to the TLS data. When detecting even smaller trees (DBH < 4 cm), in general ALS data yielded a negative Bias, on average -2.4 trees per 20 m plots. When only counting plots with N_a > 20, the average Bias was -11.6%, which can be considered a high detection accuracy.

Although ALS data yielded an under-estimation of low vegetation, it is still the preferred method for large area implementation compared with TLS data. Previous studies have proposed different solutions for

characterizing understory vegetation using ALS and TLS. Instead of investigating segmentation and positioning, most studies have estimated the coverage of the understory layer using regression models based on ALS metrics (Martinuzzi et al., 2009), with relatively sparse discrete return data (Wing et al., 2012; Morsdorf et al., 2010; Hill and Broughton, 2009) or full-waveform data (Crespo-Peremarch et al., 2018). Wing et al. (2012) and Morsdorf et al. (2010) also designed a step to classify points from different vegetation strata. In both of those studies, the classification was based on the LiDAR intensity, while we have used the morphology of the dominant trees.

The aim of this study was low vegetation segmentation, thus CR and CBH validation was not carried out. Future research could test and improve CR and CBH estimation, which would be useful for wood quality evaluation. As well as CR and CBH, other information could be calculated from the symmetry curve, e.g. tree species and crown profile. The pines and spruces showed different types of symmetry curves (funnel and cylindrical space); a preliminary classification of pines and spruces could therefore be achieved by measuring characteristics from the symmetry curve. Quantifying the diversity of forest structure is another aspect that could be developed further. At the individual-tree level, indices showing the variety of crown heights and shapes could be calculated from the symmetry curves. At the stand level, ecological indicators could be obtained, e.g. the complexity of the vertical canopy layers, the distribution of trees in horizontal space, and competition between individual trees and species. The proposed framework could also be used to generate maps indicating the ecological value of forest within large areas.

The SSD algorithm was developed based on the symmetrical structure of a single tree. We tested its performance on 251 plots in a Swedish boreal forest, with variations in stem density, vertical structure, and species composition. Studies have shown that crown asymmetry often occurs as a result of competitive pressure from surrounding trees (Seidel et al., 2011; Olivier et al., 2017). We therefore assumed that there were more asymmetrical trees in denser plots. Nevertheless, the commission error of SSD detection did not increase with increasing stem densities, and we concluded that the SSD algorithm was robust for crown asymmetry in a boreal forest. In general, deciduous trees are considered more plastic and less symmetrical than coniferous trees (Olivier et al., 2017; Martín-Sanz et al., 2016), which may lead to an under-estimate for crown diameters of dominant trees using the SSD algorithm. Some deciduous trees may not have clear treetops or have multiple treetops, which is challenging for ITC segmentation (Wang et al., 2016), especially when using algorithms involving local maxima, e.g. the SSD algorithm, k-means clustering (Gupta et al., 2010), li2012 (Li et al., 2012) and silva2016 (Silva et al., 2016) in the lidR packages. Our dataset included plots with a mix of coniferous and deciduous trees, and the accuracy of the SSD algorithm for coniferous-deciduous-mixed plots was similar to that for coniferous-only plots. We therefore suggest that the SSD algorithm is a robust application for deciduous trees in a boreal forest. We assume the SSD algorithm will work well for coniferous trees

in temperate forests and deciduous trees with a certain degree of asymmetry, but further studies are needed to test this assumption using datasets from temperate forests. The applicability of the SSD algorithm in tropical forests also needs further investigation.

6. Conclusion

This study presents a framework for ITC segmentation using laser data, to achieve better detection of low vegetation. By implementing and validating the algorithm in a boreal forest, we can draw the following conclusions. First, the new SSD algorithm, which detects the 3D symmetrical structure of individual trees to estimate the crown radius (CR) and crown base height (CBH), is an effective method for classifying points into dominant and non-dominant trees. The strength of the SSD algorithm is the consideration of the symmetry of the trees. This helps discriminate high branches from treetops and exclude points of sub-dominant trees from the dominant trees. Second, the proposed framework performs well among plots with different structures. The detection accuracy of trees with a DBH ≥ 4 cm decreased with higher stem densities and more vertical layers in the canopy; the detection accuracy of third-layer trees changed less with stem density and vertical complexity, but was influenced by tree species (ALS + TLS data). Third, the proposed framework can be used with ALS + TLS, TLS, and ALS data. Similar detection rates were achieved with TLS and ALS + TLS data, regarding detection accuracy and influencing factors. ALS data achieved lower detection rates for both second and third layers, and the accuracy was lower with higher stem densities.

The proposed framework can achieve ITC segmentation of trees in both dominant and understory layers. It can also achieve ITC segmentation in boreal forests with various complexities of structure and species composition, and from both TLS and ALS data. The SSD algorithm for detecting symmetrical structures in the trees is the key to achieving this ITC segmentation. The possible applications of the proposed framework include estimating forest structure complexity and ecological value. The SSD algorithm could be developed further, e.g. for tree species classification, and accurate estimation of CR and CBH, which could help evaluate wood quality.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgement

This research was financed by the Kempe Foundations, as part of the Finnish and Swedish bilateral research program Tandem Forest Value, and the Swedish Foundation for Strategic Environmental Research with the research program Mistra Digital Forest.

Appendix A. Abbreviation used in this study

Abbreviations and symbols	Meaning
SSD	The symmetrical structure detection algorithm
ITC	Individual tree crown
CR	Crown radius
CBH	Crown base height
T-shaped tree	A tree with a clear stem shown on the symmetry curves (Fig. 6a as an example)
A-shaped tree	A tree with a symmetrical crown to the ground (Fig. 6b as an example)
P_{dominant}	Laser points of dominant trees
P_{low}	Laser points of low vegetation
P_{all}	Laser points of the plot
P_{rest}	The rest of the laser points in a plot after removing the detected P_{dominant}

(continued on next page)

(continued)

Abbreviations and symbols	Meaning
P_{fn}	The laser points of dominant trees that were unsuccessfully classified (false negative points).
T_{CHM}	The 3D coordinates of the local maxima from the smoothed nDSM
S	Seed points
P_{Target}	The laser points that were used for testing the symmetry structure of a tree
V_{ij}	The set of the voxels of a 'crown ring'. The voxels have coordinates $h = H_i$ and $r = R_j$
(R_j, H_i)	The symmetry curve for a tree with R_j as the radius of the symmetrical crown at height H_i .
(R_u, H_u)	The crown radius R_u at the upper crown with height H_u
(R_l, H_l)	The crown radius R_l at the lower/base of the crown with height H_l
S_p	The subset of S that were determined to be true treetops

Appendix B. Parameters in the SSD algorithm and the default parameter values

Step	Parameter	Physical meaning	Default values
Step 1.1 Definition of target point clouds from individual trees	Th_{hs}	A local maximum was set as a seed if the height is higher than $Th_{hs} \times \text{Max}(\{z_k\}_{k=1}^K)$	1/3
	Th_d	A laser point was included in calculating the symmetry curve of a tree if the horizontal distance to the seed is no larger than Th_d . Set according to the largest possible crown radius	3 m
Step 1.2 Equiangular sectorial voxelization	v_d, v_z and v_α	The voxel size in the voxelization	$v_d = 0.5$ m $v_z = 0.5$ m $v_\alpha = 30^\circ$
Step 1.3 Plotting the symmetry curve	Th_v	A "crown ring" V_{ij} was determined as a symmetrical one if more than Th_v (%) of the voxels in V_{ij} were occupied.	75%
Step 1.5 Detection of CR and CBH from symmetry curves	Th_s	The value of (R_u, H_u) and (R_l, H_l) was fixed when R no longer increased more than $Th_s \times$ voxel size from H_i to H_{i+1}	1/2
Step 1.6 Creation of the clipping space	ϵ and θ	ϵ is designed to include asymmetrical branches and θ for the asymmetrical stems.	$\epsilon = 2$ voxels $\theta = 1$ voxel
Step 2.1 Mean Shift Clustering for low vegetation	Th_{hd}	Two clusters were not merged when their centers had a height difference larger than Th_{hd}	0.3
Step 2.2 Determination of true/false tree clusters	Th_z	A true tree cluster had $z_{min} < Th_z \times z_{max}$, where z_{min} and z_{max} are the minimum and maximum z values of one cluster	0.67
	Th_a	A true tree cluster had a projected area larger than Th_a	1 m ²
	Th_n	A true tree cluster had the number of points divided by the three-dimensional volume larger than Th_n	20 (ALS + TLS and TLS), 5 (ALS)

Appendix C. Sensitivity analysis of the SSD algorithm

Step	Parameter	Setting	ALS + TLS					TLS					ALS				
			DR _A	DR ₁	DR ₂	DR ₃	N*	DR _A	DR ₁	DR ₂	DR ₃	N*	DR _A	DR ₁	DR ₂	DR ₃	N*
Step 1.1 Definition of target point clouds from individual trees	Th_{hs}	1/4	0.86	0.91	0.78	0.79	21.2	0.86	0.90	0.79	0.80	20.7	0.76	0.82	0.65	0.72	17.7
		1/3	0.87	0.91	0.79	0.78	21.2	0.86	0.90	0.78	0.79	20.8	0.76	0.82	0.65	0.71	18.4
		1/2	0.86	0.91	0.76	0.78	21.1	0.86	0.90	0.79	0.79	20.8	0.76	0.83	0.65	0.70	19.0
	Th_d	2.5 m	0.85	0.88	0.78	0.80	22.6	0.85	0.88	0.78	0.81	22.4	0.76	0.81	0.66	0.72	18.7
		3 m	0.87	0.91	0.79	0.78	21.2	0.87	0.91	0.79	0.78	21.2	0.76	0.82	0.65	0.71	18.4
Step 1.2 Equiangular sectorial voxelization	v_d, v_z and v_α	0.4 m, 0.4 m, 24°	0.83	0.87	0.75	0.78	21.8	0.83	0.87	0.75	0.77	21.7	0.76	0.83	0.65	0.69	17.5
		0.4 m, 0.4 m, 30°	0.88	0.89	0.84	0.85	25.0	0.87	0.88	0.84	0.83	25.9	0.64	0.62	0.64	0.77	20.1
		0.5 m, 0.5 m, 30°	0.87	0.91	0.79	0.78	21.2	0.86	0.90	0.79	0.79	20.8	0.76	0.82	0.65	0.71	18.4
		0.6 m, 0.6 m, 36°	0.86	0.92	0.76	0.77	20.5	0.86	0.92	0.77	0.77	21.3	0.80	0.89	0.68	0.66	16.2
		0.6 m, 0.6 m, 36°	0.86	0.91	0.77	0.76	21.2	0.85	0.90	0.77	0.76	21.3	0.76	0.82	0.66	0.72	18.2
Step 1.3 Plotting the symmetry curve	Th_v	60%	0.86	0.91	0.77	0.76	21.2	0.85	0.90	0.77	0.76	21.3	0.76	0.82	0.66	0.72	18.2
		67%	0.87	0.91	0.79	0.78	21.2	0.86	0.90	0.79	0.79	20.8	0.76	0.82	0.65	0.71	18.4
		75%	0.86	0.89	0.78	0.79	22.6	0.85	0.88	0.76	0.80	22.4	0.68	0.71	0.63	0.75	19.1
Step 1.5 Detection of CR and CBH from symmetry curves	Th_s	0.4	0.86	0.89	0.77	0.80	22.4	0.85	0.88	0.78	0.81	22.5	0.68	0.71	0.63	0.75	19.3
		0.5	0.87	0.91	0.79	0.78	21.2	0.86	0.90	0.79	0.79	20.8	0.76	0.82	0.65	0.71	18.4
		0.6	0.86	0.89	0.77	0.80	22.5	0.85	0.88	0.77	0.80	22.4	0.69	0.71	0.63	0.75	19.3
Step 1.6 Creation of the clipping space	ϵ and θ	1, 0	0.89	0.92	0.83	0.83	24.9	0.88	0.90	0.83	0.84	25.0	0.70	0.72	0.64	0.76	20.0
		2, 1	0.87	0.91	0.79	0.78	21.2	0.86	0.90	0.79	0.79	20.8	0.76	0.82	0.65	0.71	18.4
		3, 2	0.85	0.89	0.75	0.75	20.1	0.84	0.88	0.76	0.77	20.3	0.68	0.71	0.62	0.72	18.2
Step 2.1 Mean Shift Clustering for low vegetation	Th_{bw}	0.8	0.86	0.91	0.78	0.76	22.9	0.85	0.90	0.77	0.78	22.7	0.75	0.82	0.63	0.69	18.1
		1	0.87	0.91	0.79	0.78	21.2	0.86	0.90	0.79	0.79	20.8	0.76	0.82	0.65	0.71	18.4
		1.2	0.85	0.91	0.76	0.72	18.6	0.84	0.90	0.76	0.74	18.4	0.77	0.83	0.67	0.72	17.2
	Th_{hd}	0.2 m	0.85	0.91	0.76	0.77	21.4	0.85	0.89	0.76	0.76	21.4	0.73	0.80	0.60	0.69	17.4
		0.3 m	0.87	0.91	0.79	0.78	21.2	0.86	0.90	0.79	0.79	20.8	0.76	0.82	0.65	0.71	18.4
0.4 m	0.86	0.91	0.76	0.76	21.1	0.85	0.90	0.77	0.76	20.9	0.78	0.84	0.67	0.73	18.5		

(continued on next page)

(continued)

Step	Parameter	Setting	ALS + TLS					TLS					ALS				
			DR _A	DR ₁	DR ₂	DR ₃	N*	DR _A	DR ₁	DR ₂	DR ₃	N*	DR _A	DR ₁	DR ₂	DR ₃	N*
Step 2.2 Determination of true/false tree clusters	Th _z	60%	0.85	0.91	0.76	0.74	20.4	0.84	0.89	0.75	0.75	20.5	0.72	0.80	0.60	0.67	16.2
		67%	0.87	0.91	0.79	0.78	21.2	0.86	0.90	0.79	0.79	20.8	0.76	0.82	0.65	0.71	18.4
		75%	0.86	0.92	0.78	0.77	21.9	0.86	0.90	0.78	0.77	21.6	0.80	0.85	0.72	0.75	20.8
	Th _a	0.8 m ²	0.86	0.91	0.76	0.77	22.3	0.85	0.90	0.78	0.77	22.1	0.76	0.82	0.65	0.73	19.6
		1 m ²	0.87	0.91	0.79	0.78	21.2	0.86	0.90	0.79	0.79	20.8	0.76	0.82	0.65	0.71	18.4
		1.2 m ²	0.85	0.91	0.76	0.74	20.1	0.85	0.90	0.76	0.75	20.1	0.75	0.82	0.64	0.71	17.2
	Th _n	18**	0.86	0.90	0.78	0.78	22.0	0.86	0.91	0.77	0.76	22.0	0.77	0.83	0.67	0.76	19.4
		6***															
		20**	0.87	0.91	0.79	0.78	21.2	0.86	0.90	0.79	0.79	20.8	0.76	0.82	0.65	0.71	18.4
		5***															
		22**	0.86	0.91	0.77	0.78	21.0	0.85	0.90	0.78	0.78	22.3	0.75	0.82	0.64	0.68	16.0
		4***															

* Number of detected low vegetation.

** Parameter values for ALS + TLS and TLS data.

*** Parameter values for ALS data.

References

- Chen, Xinxin, Jiang, Kang, Zhu, Yushi, Wang, Xiangjun, Yun, Ting, 2021. Individual tree crown segmentation directly from UAV-borne LiDAR data using the PointNet of deep learning. *Forests* 12 (2), 131. <https://doi.org/10.3390/f12020131>.
- Crespo-Peremarch, Pablo, Tompalski, Piotr, Coops, Nicholas C., Ruiz, Luis Ángel, 2018. Characterizing understory vegetation in Mediterranean forests using full-waveform airborne laser scanning data. *Remote Sens. Environ.* 217 (1), 400–413. <https://doi.org/10.1016/j.rse.2018.08.033>.
- Dai, Wenxia, Yang, Bisheng, Dong, Zhen, Shaker, Ahmed, 2018. A new method for 3D individual tree extraction using multispectral airborne LiDAR point clouds. *ISPRS J. Photogramm. Remote Sens.* 144, 400–411. <https://doi.org/10.1016/j.isprsjprs.2018.08.010>.
- Ene, Liviu, Næsset, Erik, Gobakken, Terje, 2012. Single tree detection in heterogeneous boreal forests using airborne laser scanning and area-based stem number estimates. *Int. J. Remote Sens.* 33 (16), 5171–5193. <https://doi.org/10.1080/01431161.2012.657363>.
- Ferraz, Antonio, Bretar, Frédéric, Jacquemoud, Stéphane, Gonçalves, Gil, Pereira, Luisa (Eds.), 2010. 2010 IEEE International Conference on Image Processing. 2010 17th IEEE International Conference on Image Processing (ICIP 2010). IEEE. Hong Kong, Hong Kong, 9/26/2010.
- Ferraz, António, Bretar, Frédéric, Jacquemoud, Stéphane, Gonçalves, Gil, Pereira, Luisa, Tomé, Margarida, Soares, Paula, 2012. 3-D mapping of a multi-layered Mediterranean forest using ALS data. *Remote Sens. Environ.* 121, 210–223. <https://doi.org/10.1016/j.rse.2012.01.020>.
- Gupta, Sandeep, Weinacker, Holger, Koch, Barbara, 2010. Comparative analysis of Clustering-based approaches for 3-D single tree detection using airborne fullwave lidar data. *Remote Sens.* 2 (4), 968–989. <https://doi.org/10.3390/rs2040968>.
- Harikumar, Aravind, Bovolo, Francesca, Bruzzone, Lorenzo, 2019. A local projection-based approach to individual tree detection and 3-D crown delineation in multistoried coniferous forests using high-density airborne LiDAR data. *IEEE Trans. Geosci. Remote Sens.* 57 (2), 1168–1182. <https://doi.org/10.1109/TGRS.2018.2865014>.
- Hill, R.A., Broughton, R.K., 2009. Mapping the understorey of deciduous woodland from leaf-on and leaf-off airborne LiDAR data: a case study in lowland Britain. *ISPRS J. Photogramm. Remote Sens.* 64 (2), 223–233. <https://doi.org/10.1016/j.isprsjprs.2008.12.004>.
- Holmgren, Johan, Tulldahl, Michael, Nordlöf, Jonas, Willén, Erik, Olsson, Håkan, 2019. Mobile laser scanning for estimating Tree stem diameter using segmentation and tree spine calibration. *Remote Sens.* 11 (23), 2781. <https://doi.org/10.3390/rs11232781>.
- Huo, Langning, Lindberg, Eva, 2020. Individual tree detection using template matching of multiple rasters derived from multispectral airborne laser scanning data. *Int. J. Remote Sens.* 41 (24), 9525–9544. <https://doi.org/10.1080/01431161.2020.1800127>.
- Hyypä, J., Kelle, O., Lehtikoinen, M., Inkinen, M., 2001. A segmentation-based method to retrieve stem volume estimates from 3-D tree height models produced by laser scanners. *IEEE Trans. Geosci. Remote Sens.* 39 (5), 969–975. <https://doi.org/10.1109/36.921414>.
- Hyypä, Eric, Hyypä, Juha, Hakala, Teemu, Kukko, Antero, Wulder, Michael A., White, Joanne C., et al., 2020. Under-canopy UAV laser scanning for accurate forest field measurements. *ISPRS J. Photogramm. Remote Sens.* 164 (6), 41–60. <https://doi.org/10.1016/j.isprsjprs.2020.03.021>.
- Kaartinen, Harri, Hyypä, Juha, Yu, Xiaowei, Vastaranta, Mikko, Hyypä, Hannu, Kukko, Antero, et al., 2012. An international comparison of individual tree detection and extraction using airborne laser scanning. *Remote Sens.* 4 (4), 950–974. <https://doi.org/10.3390/rs4040950>.
- Khorrami, Ramzanali, Naeimi, Zahra, Tabari, Masoud, Eslahchi, Mohammad Reza, 2018. A new method for detecting individual trees in aerial LiDAR point clouds using absolute height maxima. *Environ. Monit. Assess.* 190 (12), 708. <https://doi.org/10.1007/s10661-018-7082-8>.
- Li, Wenkai, Guo, Qinghua, Jakubowski, Marek K., Kelly, Maggi, 2012. A new method for segmenting individual trees from the lidar point cloud. *Photogramm. Eng. Remote Sens.* 78 (1), 75–84. <https://doi.org/10.14358/PERS.78.1.75>.
- Li, Jianping, Yang, Bisheng, Cong, Yangzi, Cao, Lin, Fu, Xiaoyao, Dong, Zhen, 2019. 3D forest mapping using a low-cost UAV laser scanning system: investigation and comparison. *Remote Sens.* 11 (6), 717. <https://doi.org/10.3390/rs11060717>.
- Liang, Xinlian, Hyypä, Juha, Kaartinen, Harri, Lehtomäki, Matti, Pyörälä, Jiri, Pfeifer, Norbert, et al., 2018. International benchmarking of terrestrial laser scanning approaches for forest inventories. *ISPRS J. Photogramm. Remote Sens.* 144, 137–179. <https://doi.org/10.1016/j.isprsjprs.2018.06.021>.
- Lindberg, Eva, Holmgren, Johan, Olofsson, Kenneth, Wallerman, Jörgen, Olsson, Håkan, 2013. Estimation of tree lists from airborne laser scanning using tree model clustering and k-MSN imputation. *Remote Sens.* 5 (4), 1932–1955. <https://doi.org/10.3390/rs5041932>.
- Lindberg, Eva, Eysn, Lothar, Hollaus, Markus, Holmgren, Johan, Pfeifer, Norbert, 2014. Delineation of tree crowns and tree species classification from full-waveform airborne laser scanning data using 3-D ellipsoidal clustering. *IEEE J. Sel. Top. Appl. Earth Observ. Remote Sens.* 7 (7), 3174–3181. <https://doi.org/10.1109/JSTARS.2014.2331276>.
- Ma, Zhenyu, Pang, Yong, Wang, Di, Liang, Xiaojun, Chen, Bowei, Lu, Hao, et al., 2020. Individual tree crown segmentation of a larch plantation using airborne laser scanning data based on region growing and canopy morphology features. *Remote Sens.* 12 (7), 1078. <https://doi.org/10.3390/rs12071078>.
- Ma, Kaisen, Xiong, Yujun, Jiang, Fugen, Chen, Song, Sun, Hua, 2021. A novel vegetation point cloud density tree-segmentation model for overlapping crowns using UAV LiDAR. *Remote Sens.* 13 (8), 1442. <https://doi.org/10.3390/rs13081442>.
- Martín-Sanz, Ruth C., Santos-Del-Blanco, Luis, Notivol, Eduardo, Chambel, M. Regina, San-Martín, Roberto, Climent, José, 2016. Disentangling plasticity of serotiny, a key adaptive trait in a Mediterranean conifer. *Am. J. Bot.* 103 (9), 1582–1591. <https://doi.org/10.3732/ajb.1600199>.
- Martinuzzi, Sebastián, Vierling, Lee A., Gould, William A., Falkowski, Michael J., Evans, Jeffrey S., Hudak, Andrew T., Vierling, Kerri T., 2009. Mapping snags and understory shrubs for a LiDAR-based assessment of wildlife habitat suitability. *Remote Sens. Environ.* 113 (12), 2533–2546. <https://doi.org/10.1016/j.rse.2009.07.002>.
- Matsugami, Hiroaki, 2012. Accurate detection of tree apexes in coniferous canopies from airborne scanning light detection and ranging images based on crown-extraction filtering. *Can. J. Remote Sens.* 6 (1), 63502. <https://doi.org/10.1117/1.JRS.6.063502>.
- Melzer, Thomas, 2007. Non-parametric segmentation of ALS point clouds using mean shift. *J. Appl. Geodes.* 1 (3), 6. <https://doi.org/10.1515/jag.2007.018>.
- Morsdorf, Felix, Meier, Erich H., Allgöwer, Britta, Nüesch, Daniel, 2003. *Clustering in Airborne Laser Scanning Raw Data for Segmentation of Single Trees*. In.
- Morsdorf, Felix, Märell, Anders, Koetz, Benjamin, Cassagne, Nathalie, Pimont, Francois, Rigolot, Eric, Allgöwer, Britta, 2010. Discrimination of vegetation strata in a multi-layered Mediterranean forest ecosystem using height and intensity information derived from airborne laser scanning. *Remote Sens. Environ.* 114 (7), 1403–1415. <https://doi.org/10.1016/j.rse.2010.01.023>.
- Oinas, S., Sikanen, L., 2000. Discrete event simulation model for purchasing of marked stands, timber harvesting and transportation. *Forestry* 73 (3), 283–301. <https://doi.org/10.1093/forestry/73.3.283>.
- Olivier, Martin-Ducup, Robert, Schneider, Richard, A., Fournier, 2017. A method to quantify canopy changes using multi-temporal terrestrial lidar data: tree response to surrounding gaps. *Agric. For. Meteorol.* 237–238 (4), 184–195. <https://doi.org/10.1016/j.agrformet.2017.02.016>.
- Olofsson, Kenneth, Holmgren, Johan, 2016. Single tree stem profile detection using terrestrial laser scanner data, flatness saliency features and curvature properties. *Forests* 7 (12), 207. <https://doi.org/10.3390/f7090207>.
- Paris, Claudia, Valduga, Davide, Bruzzone, Lorenzo, 2016. A hierarchical approach to three-dimensional segmentation of LiDAR data at single-tree level in a multilayered

- forest. *IEEE Trans. Geosci. Remote Sens.* 54 (7), 4190–4203. <https://doi.org/10.1109/TGRS.2016.2538203>.
- Reitberger, J., Schnörr, C., Krzystek, P., Stilla, U., 2009. 3D segmentation of single trees exploiting full waveform LIDAR data. *ISPRS J. Photogramm. Remote Sens.* 64 (6), 561–574. <https://doi.org/10.1016/j.isprsjprs.2009.04.002>.
- Seidel, Dominik, Leuschner, Christoph, Müller, Annika, Krause, Benjamin, 2011. Crown plasticity in mixed forests—Quantifying asymmetry as a measure of competition using terrestrial laser scanning. *For. Ecol. Manag.* 261 (11), 2123–2132. <https://doi.org/10.1016/j.foreco.2011.03.008>.
- Shendryk, Iurii, Broich, Mark, Tulbure, Mirela G., Alexandrov, Sergey V., 2016. Bottom-up delineation of individual trees from full-waveform airborne laser scans in a structurally complex eucalypt forest. *Remote Sens. Environ.* 173, 69–83. <https://doi.org/10.1016/j.rse.2015.11.008>.
- Silva, Carlos A., Hudak, Andrew T., Vierling, Lee A., Loudermilk, E. Louise, O'Brien, Joseph J., Hiers, J. Kevin, et al., 2016. Imputation of individual longleaf pine (*Pinus palustris* Mill.) tree attributes from field and LiDAR data. *Can. J. Remote Sens.* 42 (5), 554–573. <https://doi.org/10.1080/07038992.2016.1196582>.
- Solberg, Svein, Naesset, Erik, Bollandsas, Ole Martin, 2006. Single tree segmentation using airborne laser scanner data in a structurally heterogeneous spruce forest. *Photogramm. Eng. Remote Sens.* 72 (12), 1369–1378. <https://doi.org/10.14358/PERS.72.12.1369>.
- Tang, Feifei, Zhang, Xiaohong, Liu, Jingnan, 2007. Segmentation of tree crown model with complex structure from airborne LIDAR data. In: Ju, Weimin, Zhao, Shuhe (Eds.), *Geoinformatics 2007: Remotely Sensed Data and Information*. Geoinformatics 2007. Nanjing, China, Friday 25 May 2007: SPIE (SPIE Proceedings), 67520A.
- Tao, Shengli, Wu, Fangfang, Guo, Qinghua, Wang, Yongcai, Li, Wenkai, Xue, Baolin, et al., 2015. Segmenting tree crowns from terrestrial and mobile LiDAR data by exploring ecological theories. *ISPRS J. Photogramm. Remote Sens.* 110, 66–76. <https://doi.org/10.1016/j.isprsjprs.2015.10.007>.
- Wang, Yunsheng, Hyyppä, Juha, Liang, Xinlian, Kaartinen, Harri, Yu, Xiaowei, Lindberg, Eva, et al., 2016. International benchmarking of the individual tree detection methods for modeling 3-D canopy structure for silviculture and forest ecology using airborne laser scanning. *IEEE Trans. Geosci. Remote Sens.* 54 (9), 5011–5027. <https://doi.org/10.1109/TGRS.2016.2543225>.
- Williams, Jonathan, Schonlieb, Carola-Bibiane, Swinfield, Tom, Lee, Juheon, Cai, Xiaohao, Qie, Lan, Coomes, David A., 2020. 3D segmentation of trees through a flexible multiclass graph cut algorithm. *IEEE Trans. Geosci. Remote Sens.* 58 (2), 754–776. <https://doi.org/10.1109/TGRS.2019.2940146>.
- Windrim, Lloyd, Bryson, Mitch, 2020. Detection, segmentation, and model fitting of individual tree stems from airborne laser scanning of forests using deep learning. *Remote Sens.* 12 (9), 1469. <https://doi.org/10.3390/rs12091469>.
- Wing, Brian M., Ritchie, Martin W., Boston, Kevin, Cohen, Warren B., Gitelman, Alix, Olsen, Michael J., 2012. Prediction of understory vegetation cover with airborne lidar in an interior ponderosa pine forest. *Remote Sens. Environ.* 124 (6), 730–741. <https://doi.org/10.1016/j.rse.2012.06.024>.
- Xiao, Wen, Zafaremska, Aleksandra, Smigaj, Magdalena, Wang, Yunsheng, Gaulton, Rachel, 2019. Mean shift segmentation assessment for individual forest tree delineation from airborne lidar data. *Remote Sens.* 11 (11), 1263. <https://doi.org/10.3390/rs11111263>.
- Zhang, Junjie, Sohn, Gunho, Brédif, Mathieu, 2014. A hybrid framework for single tree detection from airborne laser scanning data: a case study in temperate mature coniferous forests in Ontario, Canada. *ISPRS J. Photogramm. Remote Sens.* 98, 44–57. <https://doi.org/10.1016/j.isprsjprs.2014.08.007>.
- Zhu, Weigang, Zhu, Chao, Zhang, Yaqiu, 2021. Research on deep learning individual tree segmentation method coupling RetinaNet and point cloud clustering. *IEEE Access* 9, 126635–126645. <https://doi.org/10.1109/ACCESS.2021.3111654>.