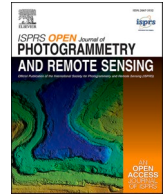




Contents lists available at ScienceDirect

ISPRS Open Journal of Photogrammetry and Remote Sensing

journal homepage: www.journals.elsevier.com/isprs-open-journal-of-photogrammetry-and-remote-sensing

Individual tree segmentation and species classification using high-density close-range multispectral laser scanning data

Aada Hakula^a, Lassi Ruoppa^a, Matti Lehtomäki^{a,*}, Xiaowei Yu^a, Antero Kukko^{a,b}, Harri Kaartinen^a, Josef Taher^{a,*}, Leena Matikainen^a, Eric Hyypä^a, Ville Luoma^c, Markus Holopainen^{c,a}, Ville Kankare^d, Juha Hyypä^{a,b}

^a Department of Remote Sensing and Photogrammetry, Finnish Geospatial Research Institute FGI, The National Land Survey of Finland, Vuorimiehentie 5, FI-02150, Espoo, Finland

^b Department of Built Environment, Aalto University, School of Engineering, P.O. Box 11000, FI-00076, Aalto, Finland

^c Department of Forest Sciences, University of Helsinki, FI-00014, Helsinki, Finland

^d School of Forest Science, University of Eastern Finland, FI-80101, Joensuu, Finland

ARTICLE INFO

Keywords:

Point cloud
Intensity
Reflectance
Mobile laser scanning
Lidar
Airborne laser scanning

ABSTRACT

Tree species characterise biodiversity, health, economic potential, and resilience of an ecosystem, for example. Tree species classification based on remote sensing data, however, is known to be a challenging task. In this paper, we study for the first time the feasibility of tree species classification using high-density point clouds collected with an airborne close-range multispectral laser scanning system – a technique that has previously proved to be capable of providing stem curve and volume accurately and rapidly for standing trees. To this end, we carried out laser scanning measurements from a helicopter on 53 forest sample plots, each with a size of 32 m × 32 m. The plots covered approximately 5500 trees in total, including Scots pine (*Pinus sylvestris* L.), Norway spruce (*Picea abies* (L.) H.Karst.), and deciduous trees such as Downy birch (*Betula pubescens* Ehrh.) and Silverbirch (*Betula pendula* Roth). The multispectral laser scanning system consisted of integrated Riegl VUX-1HA, miniVUX-3UAV, and VQ-840-G scanners (Riegl GmbH, Austria) operating at wavelengths of 1550 nm, 905 nm, and 532 nm, respectively. A new approach, *layer-by-layer segmentation*, was developed for individual tree detection and segmentation from the dense point cloud data. After individual tree segmentation, 249 features were computed for tree species classification, which was tested with approximately 3000 trees. The features described the point cloud geometry as well as single-channel and multi-channel reflectance metrics. Both feature selection and the tree species classification were conducted using the random forest method. Using the layer-by-layer segmentation algorithm, trees in the dominant and co-dominant categories were found with detection rates of 89.5% and 77.9%, respectively, whereas suppressed trees were detected with a success rate of 15.2%–42.3%, clearly improving upon the standard watershed segmentation. The overall accuracy of the tree species classification was 73.1% when using geometric features from the 1550 nm scanner data and 86.6% when combining the geometric features with reflectance information of the 1550 nm data. The use of multispectral reflectance and geometric features improved the overall classification accuracy up to 90.8%. Classification accuracies were as high as 92.7% and 93.7% for dominant and co-dominant trees, respectively.

1. Introduction

The diversity of tree species affects forest primary productivity, population resilience and recovery from disturbances, internal

competition, and fluxes of energy and nutrients. Tree species information is also an important attribute characterising ecosystem health, economic potential, resilience, and biodiversity (see e.g. Huuskonen et al., 2021; Seidl et al., 2017; Gamfeldt et al., 2013). For example, the

* Corresponding authors.

E-mail addresses: aada.hakula@gmail.com (A. Hakula), lassi.ruoppa@gmail.com (L. Ruoppa), matti.lehtomaki@nls.fi (M. Lehtomäki), xiaowei.yu@nls.fi (X. Yu), antero.kukko@nls.fi (A. Kukko), harri.kaartinen@nls.fi (H. Kaartinen), josef.taher@nls.fi (J. Taher), leena.matikainen@nls.fi (L. Matikainen), eric.hyypa@nls.fi (E. Hyypä), ville.luoma@helsinki.fi (V. Luoma), markus.holopainen@helsinki.fi (M. Holopainen), ville.kankare@uef.fi (V. Kankare), juha.hyypa@nls.fi (J. Hyypä).

<https://doi.org/10.1016/j.ophoto.2023.100039>

Received 3 March 2023; Received in revised form 11 May 2023; Accepted 15 May 2023

Available online 22 May 2023

2667-3932/© 2023 The Authors. Published by Elsevier B.V. on behalf of International Society of Photogrammetry and Remote Sensing (isprs). This is an open access article under the CC BY license (<http://creativecommons.org/licenses/by/4.0/>).

species abundance of the companion flora is particularly high for the aspen (*Populus tremula* L.), and aspen is, therefore, regarded as one of the most important species for biodiversity in some boreal forest regions (Campbell and Bartos, 2001), such as Finland. Species-specific size distribution of forest stands is also important information in forest management planning and when optimising the wood supply chain of the forest industry, thus making species as one of the most important characteristics of individual trees to be assessed. However, collecting species information with traditional field inventories covering all the trees within a specific area is not feasible due to labour cost. Remote-sensing-based solutions have been investigated intensively within the last decades (Fassnacht et al., 2016; Michalowska and Rapinski, 2021; Koenig and Höfle, 2016), but species is still one of the most challenging tree characteristics to retrieve. Therefore, there is a high demand for practical and accurate solutions for species classification of individual trees.

Today, airborne laser scanning (ALS) is used for conducting detailed forest inventories in the boreal forest zone. Holmgren and Persson (2004) demonstrated that Scots pine can be distinguished from Norway spruce with high accuracy. According to Moffiet et al. (2005), the ratio of only returns to all returns in airborne laser data separates poplar box and Cypress pine from each other. Brandtberg et al. (2003) and Brandtberg (2007) used airborne laser data under leaf-off and leaf-off/leaf-on conditions for the detection of individual trees and tree species classification. Persson et al. (2006) performed tree species classification by using both dense point clouds and multispectral images of high-resolution. Liang et al. (2007) discriminated deciduous and coniferous trees in leaf-off conditions using the difference of the first and last pulses. Koenig and Höfle (2016) summarised the lessons learnt when using geometric, waveform, and radiometric features for major species classification, focusing especially on full-waveform features.

Early attempts to use intensity data for classifying species include Ørka et al. (2009) and Korpela et al. (2010), for example. Coniferous and deciduous trees were classified with a 73% accuracy by Ørka et al. (2009), using only intensity information. In Korpela et al. (2010), an accuracy of 88%–90% was achieved for separation between pine, spruce, and birch trees using well-calibrated intensity data. They also concluded that species such as *Salix caprea*, *Alnus incana*, and *Alnus glutinosa*, which are ecologically important and rare in Finland, could be separated using their high intensity values. Since then, multispectral airborne laser scanning using the Optech Titan sensor (Teledyne Optech, Ontario, Canada), providing intensity information from three wavelength channels and the corresponding point clouds, has offered advanced means for species classification. In Yu et al. (2017), the classification of pines, spruces, and birches was tested using Titan data and four sets of features, namely 1) features using geometry, 2) features using intensity from three wavelengths, 3) features using geometry and intensity from one channel, and 4) features using geometry and intensity from three wavelengths. The corresponding classification accuracies were 76%, 85%, 82%, and 86%, respectively. Thus, the geometric features did not provide any added value compared to the intensity features from the three channels. There have also been studies where multispectral Optech Titan data have been compared with image-matching based solutions. In Kukkonen et al. (2019a) it is claimed that ALS integrated with optical passive imagery features performed better than multispectral Titan ALS data alone.

In future, automated techniques for species classification are also needed for the automated forest field reference data collection. Today remote sensing-assisted forest inventories are based on airborne or spaceborne remote sensing data, covering the studied forest areas, and on field reference plots, collected manually and representing the variability of key forest attributes. The key forest attributes include species, biomass, volume, basal area, and height, for example. In future, forest field reference data could be collected using laser scanners installed on a drone or a helicopter, as suggested by Hyyppä et al. (2022). They showed that sophisticated point cloud processing algorithms enable one

to measure the stem curves and volumes of individual trees directly from high-density point cloud data collected using close-range ALS from a drone or from a helicopter. They concluded that the root-mean-square errors of stem curve and volume estimates were approximately 2 cm and 10%–15%, respectively, and that the technology has the potential for collecting data of several hundred reference trees per minute, which is two orders of magnitude faster compared to methods based on terrestrial or under-canopy mobile laser scanners. Since the accuracy of stem attribute estimation was almost comparable to terrestrial mobile laser scanning surveys (Hyyppä et al., 2020b), Hyyppä et al. (2022) suggested that close-range airborne laser scanning is a potential candidate for providing field reference for future individual-tree level inventories with a high efficiency. In order to provide such field reference, the measurement and analysis methods in Hyyppä et al. (2022) must, however, be extended to also accurately classify the tree species. Hyyppä et al. (2022) used single scanner data from Riegl VUX-1HA scanner (Riegl GmbH, Austria) operating at a wavelength of 1550 nm.

In our study, the system of Hyyppä et al. (2022) was extended with additional scanners operating at wavelengths of 905 nm and 532 nm, in order to provide more spectral and geometric information on individual trees. The feasibility of this new close-range, multispectral laser scanning system, mounted on a helicopter, was studied for tree species classification in boreal forest conditions. This is the first time, when high-density (100+ pts/m²) point clouds with multispectral reflectance information were applied for tree species classification. In addition, a new approach was developed for individual tree detection and segmentation from the dense point cloud data. To study the advantages of the multispectral approach, classification tests were carried out using geometric features alone and combined with reflectance features from one or several channels. Furthermore, tree detection and species classification were analysed in crown categories ranging from suppressed to dominant trees.

2. Materials

2.1. Test site and characteristics

Our test site, known as Scan Forest, is located in the boreal forest zone near Evo, Finland (61.19°N, 25.11°E), and it is a Finnish research infrastructure for forest studies. In this work, we utilised 53 test plots with a size of 32-m-by-32-m. Majority of the trees were either Scots pines (*Pinus sylvestris* L.), Norway spruces (*Picea abies* (L.) H.Karst.), Silverbirches (*Betula pendula* Roth), or Downy birches (*Betula pubescens* Ehrh.). Several other deciduous species were also present, but they only accounted for a minor portion of all the trees. In the field reference data, the following information was available for each tree within the plots: position, species, height, diameter, and volume. The positions of trees were determined from maps that were created using terrestrial laser scanning and Global Navigation Satellite System (GNSS) measurements. The tree species were identified interactively in the field and from the 3D data. Tree heights were measured with a clinometer, diameters at breast height (DBHs) with a caliper, and volumes were estimated using national allometric models based on the measured DBH and height.

The descriptive statistics of the 53 sample plots consisting in total of more than 5000 trees used for the study are summarised in Table 1.

Table 1

Descriptive statistics of the 53 reference plots. SD denotes the standard deviation of the corresponding metric.

	Minimum	Maximum	Mean	SD
Lorey's height (m)	14.8	31.3	22.3	4.3
Basal area weighted mean DBH (cm)	15.8	46.4	27.5	8.3
Basal area (m ² /ha)	15.8	50.7	30.4	8.6
Tree volume (m ³ /ha)	130.7	550.9	318.8	120.7
Total biomass (Mg/ha)	68.4	285.2	156.5	52.4
Density (trees/ha)	341.8	2998.1	1022.4	616.8

Table 2

Species-wise descriptive statistics of the reference trees. The number in brackets specifies the number of dead trees among the trees of the corresponding species. SD denotes the standard deviation of the corresponding metric.

		Minimum	Maximum	Mean	SD	Number of trees
Pine	Tree height (m)	1.8	37.2	18.9	5.1	1402 (55)
	DBH (cm)	5.0	66.7	21.3	8.9	
Spruce	Tree height (m)	1.5	38.2	14.3	8.2	2447 (93)
	DBH (cm)	5.0	73.2	15.8	10.5	
Deciduous	Tree height (m)	1.7	34.2	16.9	5.8	1700 (89)
	DBH (cm)	4.6	55.4	14.3	7.2	

Table 3

Definitions of the four distinct tree crown categories and the number of trees in each category. Two trees are considered neighbours if the distance between them is less than 3 m.

Category	Number of trees	Description	Definition
A	1375	Isolated or dominant trees	No neighbouring trees or ≥ 2 m higher than all neighbours.
B	1303	Group of similar trees	At least one neighbouring tree and the height difference to all neighbours is < 2 m.
C	1450	Tree alongside a dominant tree	At least one neighbouring tree is ≥ 2 m higher and at a distance of ≥ 1.5 m.
D	1421	Tree under a dominant tree	At least one neighbouring tree is ≥ 2 m higher and at a distance of < 1.5 m.

Similarly, a summary of the species-wise descriptive statistics of the reference trees are presented in [Table 2](#).

We separated the reference trees into four distinct crown categories based on distance and height difference in comparison to the neighbouring trees. The employed category descriptions are equivalent to those used in [Yu et al. \(2017\)](#). In our analysis, two trees were considered neighbours if the distance between them was less than 3 m. The trees were divided into categories to analyse how the detection and classification accuracies are affected by the relative crown position with respect to the neighbouring trees. The formal category definitions along with the number of trees in each category are presented in [Table 3](#). The category labels were assigned to the reference trees using an automated algorithm. Consequently, some trees may have been incorrectly classified to a more dominant category on the edges of the reference data coverage, where information about all neighbouring trees was not available.

2.2. System used and data collection

The data collection campaign was launched on 22 June 2021 in leaf-on conditions, when the intensity of deciduous trees is close to their maximum level at 1550 nm, based on LiPhe terrestrial monitoring station ([Campos et al., 2021](#)). By 22 June, leaves had reached their maximum size and coniferous trees their annual growth. The multispectral laser scanning data were collected using Finnish Geospatial Research Institute's in-house developed laser scanner system called HeliALS-TW, equipped with three Riegl scanners (Riegl GmbH, Austria), namely, VUX-1HA (scanner 1), miniVUX-3UAV (scanner 2), and VQ-840-G (scanner 3) with wavelengths of 1550 nm, 905 nm, and 532

nm, respectively. The positioning system consisted of NovAtel (LITEF) ISA-100C inertial measurement unit, a NovAtel PwrPak7 GNSS receiver, and a NovAtel (Vexxis) GNSS-850 antenna. For this study, the system was mounted to a helicopter, and multispectral data were collected by flying over the test sites from two perpendicular directions at 80 m above ground level. To maximise the number of photons hitting the tree trunks, instead of just the canopy, the scan planes were tilted 15° forwards with respect to the vertical plane. The trajectory of the laser scanning system was computed using Waypoint Inertial Explorer (version 8.9, NovAtel Inc., Canada) and a single virtual GNSS base station from Trimnet service (RINEX 3.04), positioned roughly in the middle of the site. [Table 4](#) contains detailed information about the characteristics of the data and specifications of the systems. Point density was defined as the number of points per square metre, which in our study was higher than the number of sent pulses due to multiple returns in forested areas.

3. Methods

3.1. Developed tree retrieval algorithms

In this section we propose a novel tree segmentation algorithm, *layer-by-layer segmentation*, which is designed for segmenting forest plots of moderate density with a substantial number of understory trees that are difficult to detect from above the canopy. The algorithm can detect trees whose stems are either visible or occluded in the point cloud, including small trees below the canopy. The tree detection phase of the layer-by-layer segmentation, presented in [Section 3.1.1](#), is largely based on the algorithm proposed by [Hyyppä et al. \(2020a\)](#) that does not require that

Table 4

The specifications of the laser scanners in the multispectral system used in the study. Note that scanner 2 has an elliptical beam. For scanner 3, the maximum scanning angle is separated into two parts, and expressed as the angle in flight direction \times the angle perpendicular to flight direction.

Scanner	1	2	3
Model	VUX-1HA	miniVUX-3UAV	VQ-840-G
Wavelength (nm)	1,550	905	532
Flight altitude (m)	80	80	80
Approx. point density (points/m ²)	1,400	500	1,600
Number of returns	12	5	15
Maximum scanning angle (°)	360	120	28 \times 40
Laser beam divergence (mrad)	0.5	0.5 \times 1.6	1
Laser beam diameter at the ground level (cm)	4	4 \times 12.8	8
Pulse repetition rate (kHz)	1,017	300	200

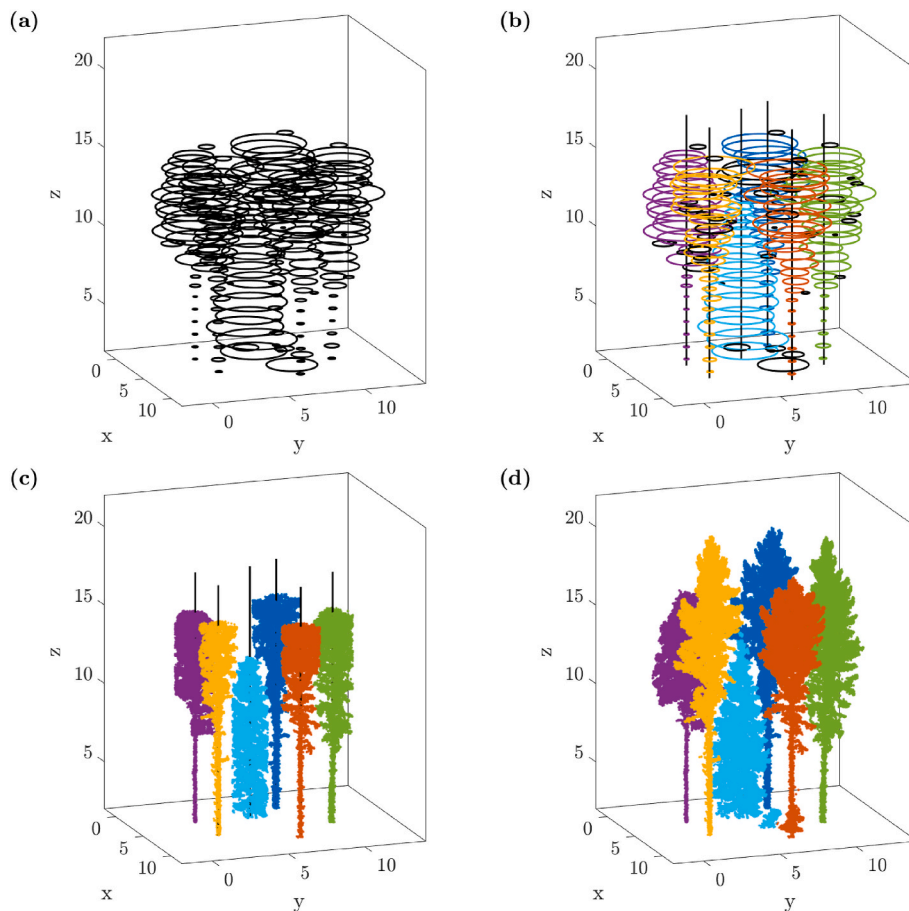


Fig. 1. Illustration of the operating principle of the layer-by-layer segmentation algorithm. (a) The circular representations of the DBSCAN clusters (Ester et al., 1996) formed in the initial step. Each cluster is drawn on the vertical centre of the layer it belongs to. (b) Vertical lines obtained by applying the line fitting algorithm on the set of clusters from (a). Clusters assigned to the same vertical line are coloured with the same colour, while clusters that could not be assigned to any line are left black. (c) Augmented training data constructed based on the clusters assigned to the vertical lines. (d) Segments formed by iteratively classifying the points in each layer using an FkNN classifier trained with the training data from (c) and previous layers. (For interpretation of the references to colour in this figure legend, the reader is referred to the Web version of this article.)

the tree trunk is visible in the point cloud. For each detected tree, the algorithm outputs a set of corresponding points, which are then used as the input for the segmentation phase presented in Section 3.1.2. The segmentation phase relies on iteratively classifying the points in each layer using a Fuzzy k-Nearest Neighbours (FkNN) classifier. Fig. 1 illustrates the operating principle of the algorithm. All parameter values utilised in the tree segmentation process were chosen based on trial and error, using two forest plots of moderate density as a point of reference. The segmentation algorithm was implemented with the Python programming language (Van Rossum and Drake, 2009). The library scikit-learn (Pedregosa et al., 2011) was utilised for clustering and classification, while other data processing was done using the libraries NumPy (Harris et al., 2020) and pandas (The pandas development team, 2020; Wes McKinney, 2010).

3.1.1. Initial tree detection

Point cloud normalization. Prior to tree detection, we normalised the ALS point cloud by subtracting the ground elevation from the z-coordinates using a digital terrain model (DTM). To construct the DTM, we divided the point cloud into a rectangular grid in the horizontal plane and retrieved the lowest point from each grid cell using the method of Hyyppä et al. (2020b) without Gaussian smoothing. Subsequently, we extracted smooth surfaces from the lowest points utilising the surface-growing-based ground extraction algorithm of Lehtomäki et al. (2016) and reconstructed the DTM from the obtained surface patches.

DBSCAN clustering. In the initial step of the tree detection algorithm, the normalised point cloud was divided into horizontal layers. The heights of the layers were set to 0.8 m, 0.6 m, and 0.4 m for the intervals [2 m, 6 m], [6 m, 10 m], and [10 m, ∞ m], respectively. A greater layer height was chosen for lower sections of the point cloud to account for reduced point density.

To identify potential locations of individual trees, we projected the points onto the xy-plane and applied the DBSCAN clustering algorithm (Ester et al., 1996) to each layer separately. Only layers between the heights of 2 and 15 m were clustered, which comprises a total of 25 layers. In the clustering, the value chosen for the neighbourhood radius was $\epsilon = 0.3$ m, and the threshold for the minimum number of points in a neighbourhood was set to $\min_{\text{pts}} = 16$ for layers above the height of $h = 10$ m and $\min_{\text{pts}} = 10$ for the layers below. For each of the resulting clusters, we created a corresponding circular representation. The centre of the circle was defined as the arithmetic mean of the cluster, and the effective radius was determined using the following equation

$$r_{\text{eff}} = \min\{d_{\text{max}}, 2.5 \text{ m}\}, \quad (1)$$

where d_{max} is the maximum distance from the cluster centre to the set of all points in the cluster. Clusters where $d_{\text{max}} > 2.5$ m frequently contain points from multiple trees, thus restricting the effective radius was necessary to minimise the adverse effects of multi-tree clusters at later stages of the segmentation process.

Local maximum detection. On lower layers of dense forest plots, the DBSCAN algorithm frequently fails to find clusters that correspond to existing trees, as the number of points tends to be quite low. To alleviate this problem for dominant and co-dominant trees (crown categories A and B), we added simple local maximum detection to our algorithm. Local maxima are a popular choice for tree detection (see e.g. Dalponte and Coomes, 2016; Silva et al., 2016; Yang et al., 2020), as they tend to correspond to existing tree locations reasonably accurately. While local maxima do not circumvent the problem for suppressed trees in categories C and D, they provide a clear improvement in the detection rate of trees in categories A and B on dense forest plots, as the maxima can compensate for inaccurate DBSCAN clusters formed on lower layers.

We utilised the local maxima detection implementation proposed by Yu et al. (2017). As a first step, a rectangular grid with cell width of 0.3 m was created, and each point in the point cloud was assigned to a cell based on its location. Next, we constructed a canopy height model (CHM) by setting the value of each cell to the maximum z -coordinate of all first returns within the cell. Bilinear interpolation was utilised to find a value for the cells that contained no points, and all cells with a value below 2 m were classified as background. Subsequently, noise was removed on the surface of tree crowns by smoothing the corrected CHM using a Gaussian filter. The standard deviation and window size used for smoothing were set to $\sigma = 0.7$ pixels and 5 pixels \times 5 pixels, respectively.

To detect the local maxima, we first applied a maximum filter to the smoothed CHM with a window size of 5 pixels \times 5 pixels. Any cells in the maximum filtered CHM for which the value remained equal to the corresponding cell in the smoothed CHM were then considered local maxima. A circular representation was constructed for each detected maximum similarly as for the clusters retrieved using DBSCAN, however, now setting the effective radius to $r_{\text{eff}} = 0.5$ m. The representations were subsequently appended to the set of DBSCAN clusters on a separate layer above the layers created in the previous step.

Tree location detection. Having performed the local maxima detection, a vertical line fitting algorithm was applied on the extended set of all clusters. The aim of this step was to detect the locations of individual trees. The line fitting procedure comprised the following steps:

1. For each cluster, fit a vertical line through the cluster centre, then find all inliers by retrieving clusters with a centre within half of their effective radius r_{eff} of said line. Assign at most one inlier from each layer to the line. If there are multiple inliers from the same layer, the cluster nearest to the line is chosen. From the set of lines fitted to the cluster centres, select the line with the most inlying clusters assigned to it and discard others.
2. Accept the line if at least 8 inlying clusters are assigned to it or if the line is associated with at least 5 inlying clusters and all clusters are located below the 11th layer (9.6 m). Remove the inlying clusters from the list of available clusters.
3. Determine the location of the accepted line by computing the weighted average of the centres (x_i, y_i) of the clusters assigned to the line using equations

$$x = \frac{\sum_i x_i r_{\text{eff},i}^2}{\sum_i r_{\text{eff},i}^2}, \quad (2)$$

$$y = \frac{\sum_i y_i r_{\text{eff},i}^2}{\sum_i r_{\text{eff},i}^2}, \quad (3)$$

where i denotes the index and $r_{\text{eff},i}$ the effective radius of the cluster.

4. Find all previously accepted lines that are within a distance of 0.5 m from the newly accepted line. If any exist, delete the closest line from the list of known lines and assign all clusters associated with it to the new line. Set the location of the new line to the mean of the two lines. Repeat this process until the distance from the new line to the nearest line is greater than 0.5 m. Based on our tests, the threshold value of 0.5 m merges a majority of the fitted vertical lines that belong to the same tree, while simultaneously not merging lines representing adjacent trees in most cases.
5. Repeat the described process for the set of remaining clusters from step 2, until the highest number of inlying clusters assigned to a line is less than 5.

Following the initial line fitting procedure, we iterated through the fitted vertical lines and retrieved all clusters with a centre within a distance of $0.75r_{\text{eff}}$. Subsequently, the retrieved clusters were assigned to the corresponding vertical line. This step was necessary, since each cluster could be assigned to at most one vertical line in the initial line fitting procedure, even though clusters may contain points from more

than one tree. Furthermore, the number of clusters that could be assigned to a vertical line was limited to one from each layer, while multiple clusters from the same layer may in fact be part of the same tree. As a consequence of reassigning the clusters, a single cluster could be associated with multiple vertical lines. All such clusters were split, such that each point in the cluster was assigned to the nearest associated vertical line with respect to the weighted distance in the xy -plane $\|\mathbf{p}_i - \mathbf{l}_j\|_2 / r_{\text{eff}}$, where $\|\cdot\|_2$ denotes the Euclidean distance, and \mathbf{p}_i and \mathbf{l}_j denote the coordinate vectors of the i th point in the cluster and j th associated line, respectively. Having performed the cluster splitting, we recomputed the locations of the vertical lines using the newly assigned clusters and Equations 2 and 3.

Post-processing of output point clouds. As a final step of the tree detection process, we looped through the vertical lines once more and retrieved all large clusters, that is, clusters for which the effective radius had been set to the maximum value $\max\{r_{\text{eff}}\} = 2.5$. For each large cluster, we computed the distance from each point to the corresponding vertical line. Subsequently, all points for which the distance was greater than $\max\{r_{\text{eff}}\}$ were discarded. The objective of this post-processing step was to remove any points belonging to adjacent trees from the large clusters, since such points could result in significant number of erroneous point classifications when forming the final tree segments.

3.1.2. Tree segmentation

The final tree segments were formed using a Fuzzy k-Nearest Neighbours (FkNN) classifier (Keller et al., 1985) iteratively layer by layer. As the name suggests, FkNN is a fuzzy variant of the traditional k-Nearest Neighbours algorithm (kNN). For each input, the classifier returns a vector containing the probability estimates of each known class label, based on the most common classes among the k nearest neighbours of the input, relative to a given distance metric. To train the classifier, we constructed training data from the points assigned to the detected trees, that is, the output of the tree detection phase.

As a first step, we cut a cylinder from the point cloud of each tree, using the location \mathbf{l}_i of the corresponding vertical line as the centre, where i is the index of the line. The radius was set to

$$r_c = 0.5 \max_{\mathbf{p} \in \mathbf{P}_i} \{\|\mathbf{p} - \mathbf{l}_i\|_2\}, \quad (4)$$

where \mathbf{P}_i is the set of points assigned to the i th vertical line. The purpose of this step was to minimise the number of points from adjacent trees in the training data.

Next, we searched for vertical gaps from each cylindrical point cloud separately. First, the points were sorted in an ascending order with respect to the z -coordinate. Subsequently, a location in the sorted list was classified as a gap if the difference in z -coordinates z_{diff} between two consecutive elements was more than 0.3 m. The gaps for which $z_{\text{diff}} \leq 1$ m were classified as short gaps, and other gaps were classified long. As an exception, if the lowest layer of the point cloud contained no points, the resulting gap created by the first layer with points was categorised short regardless of its length.

If one or several large gaps were found from a cylinder during the search, all points above the lowest such gap were removed from the cylinder. This was necessary, since the presence of a large gap frequently implies that the upper clusters belong to a dominant tree adjacent to a smaller tree, to which the lower clusters belong. Following the removal of large gaps, any remaining short gaps were populated using data augmentation. In the data augmentation, we first cut a slice of thickness 0.4 m from the point cloud both above and below the gap. Subsequently, we sampled n_{draws} points with replacement from the slices. The sample size was set to $n_{\text{draws}} = n_{\text{points}}/0.8$, where n_{points} is the number of points in the slices combined. Lastly, we altered the z -coordinate of the sampled points by drawing a new value from a uniform distribution that spans the height of the gap. Finally, each detected tree was assigned a unique integer label, and the points in the training data were labelled

correspondingly.

Subsequently, the original point cloud of the forest plot and the newly constructed point cloud training data were divided into equilateral voxels of width $w_{xyz} = 0.3$ m. All voxels that contained at least one point were classified as full. To construct a voxelised version of the training data, each voxel that contained points from the point cloud training data was assigned the corresponding integer label. We then initialised an FkNN classifier, and trained it using the voxelised training data, with the x -, y -, and z -coordinates of the voxels as features. Consequently, for any given voxel, the output of the classifier was a vector containing the probability estimates of belonging to each detected tree. The number of neighbours was set to $k = 3$, and the metric used for distance computation was the three-dimensional Euclidean distance.

Having trained the classifier, we iterated through each layer in the voxelised point cloud, starting from the lowest. On each iteration, we first computed a probability estimate of each label for every full voxel in the layer using the FkNN classifier. When computing the estimates, the voxels in each neighbourhood were weighted by the inverse of their distance from the query point. Each voxel was then assigned the most probable label, provided that the probability estimate was above or equal to the threshold value $p_{\min} = 0.9$. Voxels with no such label were discarded, and the labelled voxels were appended to the training data. Finally, the FkNN classifier was retrained using the latest training data, before proceeding to the subsequent layer. Setting a probability threshold for voxel labelling was necessary to minimise segment spilling, where misclassifying a voxel in lower layers causes increasingly detrimental misclassifications in upper layers, such that the segment begins spreading to adjacent trees. Based on our observations, segment spilling was considerably more common if a traditional kNN classifier was applied in place of the FkNN classifier with a probability threshold.

As a final step, we performed a simple outlier detection on each tree segment to remove any points that were improbably part of the particular tree. Each segment was processed in slices that were 4 m long in the vertical direction. First, we sorted the points in an ascending order based on their horizontal distance from the tree location. If the difference in distances between two consecutive entries of the list was greater than 0.3 m, all subsequent points were considered outliers and removed from the segment.

3.1.3. Reference segmentation method

To assess the performance of the layer-by-layer segmentation algorithm, we tested its efficacy against a popular individual tree segmentation algorithm based on watershed delineation of individual tree crowns. The algorithm comprised the following steps.

1. Create a CHM of the point cloud, and detect local maxima, for example, using the method presented in Section 3.1.1. These local maxima are considered as tree tops.
2. Using the tree tops as markers, perform a marker-controlled watershed transformation on the CHM to delineate individual tree crowns.
3. For each crown segment, retrieve all points under the area it covers to form the individual tree segments.

While, many implementations of the algorithm in question exist (see e.g. [Chen, 2007](#); [Edson and Wing, 2011](#); [Roussel et al., 2020](#); [Roussel and Auty, 2022](#)), we chose to use the algorithm of [Yu et al. \(2011\)](#). For the local maxima detection phase of the algorithm, the same parameters as in Section 3.1.1 were used.

3.1.4. Matching tree segments to reference data

The tree segments were matched to field-measured reference data using the Hausdorff-distance-based matching algorithm proposed by [Yu et al. \(2006\)](#), which uses the three-dimensional Euclidean distance between the tree segments and reference measurements as a matching

criterion. The distance was computed using the positions and heights of the segment and reference trees. A tree segment **A** and a reference tree **B** were considered a match if **B** was the closest to **A** out of all reference trees, and similarly **A** was the closest to **B** out of all tree segments, provided that the distance between the two was below the threshold $d_t = 5$ m. The threshold distance was set to 5 m to account for the tendency for tree height underestimation in reference measurements and differences caused by the method of determining the horizontal location between the segmentation algorithm and reference collecting procedure. We chose to use the same reference matching algorithm and threshold distance as [Yu et al. \(2017\)](#), to make our results directly comparable.

Segments that were successfully matched to a reference tree were considered true positives, while segments with no match were false positives. Similarly, reference trees for which no matching segment was found resulted in a false negative. To assess the detection accuracy of layer-by-layer segmentation, we used the metrics precision, recall, and F_1 -score. The former two are also frequently referred to as correctness and completeness, respectively, in remote sensing literature. If the number of true positive items equals TP, the number of false positive items FP, and the number of false negative items FN, the metrics can be computed as follows:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \quad (5)$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \quad (6)$$

$$F_1 = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \quad (7)$$

3.2. Developed tree species classification algorithms

This section presents the background of the developed tree species classification algorithm. We begin by introducing the derived features, after which we discuss the feature selection, and the classification of the tree species.

3.2.1. Derived features

The features computed were divided into three categories: geometric features, single-channel reflectance (SCR) features, and multi-channel reflectance (MCR) features. In total, 249 features were computed. The number of features in the three above categories were 34, 147, and 68, respectively.

Geometric features contained the standard deviation (H_{std}), mean (H_{mean}), and maximum (H_{max}) of the normalised heights, height percentiles at 5% and 99%, and from 10% to 90% with a 10-percentage-point increment (HP_j , $j = 5, 10, 20, 30, \dots, 90, 99$), crown area (CA), volume (CV), and diameter (CD), and the ratio of the number points in the lowest 2 m to the total number of points (P). In addition, we computed ratios of height sections D_1, \dots, D_{10} by dividing the z -axis into ten equal sections from the ground level up to the tree height, and by computing the ratio of the number of points in each section to the total number of points in the tree segment.

Geometric features also contained four echo features D_{only} , D_{first} , $D_{\text{intermediate}}$, and D_{last} , which were the ratios of only, first, intermediate, and last returns, respectively, to the total number of points in a tree segment. Moreover, the mean ($D_{\text{last,mean}}$), and the standard deviation ($D_{\text{last,std}}$) of the normalised heights of the last returns were computed as well.

SCR features were computed for each laser scanner separately. These features included minimum, maximum, range, mean, standard deviation, skewness, kurtosis, and histogram of the reflectance as well as the reflectance percentiles at 5%, and from 10% to 90% with a 10-percentage-point increment. The single-channel reflectance histograms were calculated using 32 bins. We used a normalised histogram for each tree, that is, we divided the bin heights by the sum of the bin heights. The

Table 5

Classification features extracted from tree segments. Superscripts correspond to the scanner numbers 1, 2, and 3 used to obtain the SCR feature. The subscript F corresponds to the SCR feature used to compute the MCR feature. Detailed descriptions of the features can be found in the text.

Feature definition	Symbol/formula
<i>Geometric features</i>	
Maximum height	H_{\max}
Mean height	H_{mean}
Standard deviation of height	H_{std}
Crown area	CA
Crown volume	CV
Crown diameter	CD
Ratio of the points below 2 m	P
Height percentiles, $j = 5, 10, 20, 30, \dots, 90, 99$	HP_j
Ratios of height sections, $k = 1, \dots, 10$	D_k
Ratio of only returns	D_{only}
Ratio of first returns	D_{first}
Ratio of intermediate returns	$D_{\text{intermediate}}$
Ratio of last returns	D_{last}
Mean height of the last returns	$D_{\text{last, mean}}$
Standard deviation of height of the last returns	$D_{\text{last, std}}$
<i>Single-channel reflectance features</i>	
Minimum reflectance	I_{\min}^i
Maximum reflectance	I_{\max}^i
Mean reflectance	I_{mean}^i
Standard deviation of reflectance	I_{std}^i
Skewness of reflectance	I_{skew}^i
Kurtosis of reflectance	I_{kurt}^i
Range of reflectance	I_{range}^i
Reflectance percentiles, $l = 5, 10, 20, 30, \dots, 90$	$I_{\text{prct, } l}^i$
Reflectance histograms, $m = 1, 2, \dots, 32$	$I_{\text{hist, } m}^i$
<i>Multi-channel reflectance features</i>	
Ratios of reflectance features	$R_F^i = I_F^1 / (I_F^1 + I_F^2 + I_F^3)$
Index of reflectance features	$N_F = (I_F^2 - I_F^3) / (I_F^2 + I_F^3)$

denominator equalled the number of points in the tree. The normalised histogram approximated the probability density function of the reflectance and was invariant to tree size. The bin heights of the normalised histograms were then used as features.

MCR features were the ratios and indices of SCR features, excluding histograms. Reflectance ratio was computed by dividing the SCR feature of one system by the sum of the corresponding SCR features of all systems. The reflectance index for each SCR feature was computed as the difference of scanner 2 (905 nm) and scanner 3 (532 nm) features divided by their sum. Corresponding wavelengths have been used in previous studies to calculate pseudo NDVI (pseudo normalised difference vegetation index) from multispectral ALS data (Wichmann et al., 2015).

Finally, all features were normalised by using the standardised z-scores, after which features' means equalled zero and standard deviations equalled one. Features and their definitions can be found in Table 5.

3.2.2. The selection of features and classification of tree species

In the tree species classification, the trees were categorised into three classes, namely, pines, spruces, and deciduous trees, using random forest (Breiman, 2001). We compared four classification models corresponding to feature subsets, which are described in Section 4.2. Trees were randomly divided into training and test sets, which were equally sized, and the ratios of the trees of different species and crown categories were similar as well. The training set was used to train the models and for feature selection, while the performances of the models were evaluated using the test set. We only considered true positive tree segments

obtained with the tree segmentation methods presented in Section 3.1. All tree segments with less than 1500 points were removed, so the total number of trees used in tree species classification was 2948, and the sizes of the training and test set were 1473 trees, and 1475 trees, respectively.

For each classification model, we performed feature selection to remove features that could degrade the performance of the classifier. Feature selection was conducted using the out-of-bag predictor importance values by permutation (Mathworks, 2022), using the implementations in Matlab R2022a. In this method, the importance of a feature is roughly defined in the following way. If a feature is important, then permuting its value should affect the performance of the model, whereas if a feature is not important, then leaving it out or permuting its value should have very little effect on the performance of the model. In this study, the optimum number of features was selected based on trial and error. The selection was performed separately for each model.

When training random forest classification models, we formed one hundred decision trees, and the number of randomly selected predictors at each decision split was 3, 4, 4, and 6 for models 1 through 4, respectively. These were the default values in Matlab, and based on our tests, the random forest models were robust against variations in these values.

For evaluating the performance of the models, we computed the overall accuracies. For analysing the misclassification of the species, the confusion matrices were computed as well, and using the confusion matrices, we calculated the corresponding kappa coefficients. Kappa coefficient, also known as Cohen's kappa, compares the accuracy that was observed to the accuracy that was expected, that is, it measures whether the conducted classification was better than a random chance. We also computed the class-wise precisions and recalls, using Equations 5 and 6, respectively. In the species classification, a tree was considered true positive when it was classified correctly. If a tree was falsely classified into class $c \in \{\text{pine, spruce, deciduous}\}$, the tree was considered false positive for that class. If a tree in class c was falsely classified into some other class, the tree was considered false negative when computing the recall of class c .

4. Results and discussion

4.1. Results for detection of trees

Results of the tree detection in the form of recall, precision, and F_1 -score for both the layer-by-layer segmentation and the watershed delineation are presented in Table 6. Overall, out of the 5549 reference trees, the layer-by-layer segmentation correctly detected a total of 3074 trees, a 23% increase compared to the 2495 trees detected using the watershed delineation. At plot level, the results of both algorithms fluctuated significantly depending on the plot type. As such, we included

Table 6

Number of trees detected by the layer-by-layer segmentation and the watershed delineation, along with the respective recall, precision, and F_1 -score. To account for high variation in results between different plot types, both the overall and average value of each accuracy metric are included. The overall metrics are computed using the set of all trees from all plots, while the average metrics are the mean of the corresponding plot-wise metrics across all plots.

	Layer-by-layer segmentation		Watershed delineation	
	Overall	Average	Overall	Average
Number of trees measured	5549		5549	
Total trees segmented	3838		2982	
Correctly detected	3074		2495	
	Overall	Average	Overall	Average
Precision	0.801	0.800	0.837	0.837
Recall	0.554	0.655	0.450	0.544
F_1 -score	0.655	0.702	0.585	0.643

both the overall and average value of each accuracy metric in Table 6. The overall metrics measure accuracy on the set of all trees across all plots, while average refers to the mean of the plot-wise values of the corresponding metric. For the layer-by-layer segmentation, the plot level precision, recall, and F_1 -score varied between 0.577 and 1, 0.241 and 0.969, and 0.349 and 0.961, respectively. The corresponding ranges for the watershed delineation were [0.5, 1], [0.128, 0.918], and [0.206, 0.933]. These results suggest, that even in the worst case scenario the overall performance of the layer-by-layer segmentation generally provides an improvement over the watershed delineation. Conversely, in the best case scenario the accuracy metrics of the two algorithms are comparable, although the layer-by-layer segmentation appears to perform slightly better. The aforementioned observations are supported by the overall accuracy metrics in Table 6: the precision of the layer-by-layer segmentation and the watershed delineation are approximately equal at 0.801 and 0.837, respectively, while the difference in recall is evident, with the former algorithm achieving an overall value of 0.544 and the latter 0.450. Although our results suggest that the performance of the layer-by-layer segmentation is better on average, this does not imply that the watershed delineation is guaranteed to perform worse at all times. In fact, on some of our denser test plots we achieved a better detection rate using the watershed delineation.

Aside from very sparse forest plots of trees of similar height, which are trivial to segment even with the watershed delineation, the layer-by-layer segmentation performed best on plots of trees with varying height and with density ranging from sparse to moderately dense. An example of the results of individual tree segmentation for a moderately dense plot is presented in Fig. 2. Conversely, the lowest detection accuracies were observed on extremely dense plots, especially those with trees of varying height. On dense plots, the clusters formed from the lower layers of the point cloud often contain multiple adjacent trees, since the point density near the ground is low compared to sparse and moderately dense plots. Consequently, the detected tree locations can be rather inaccurate due to merging of adjacent trees, which generally results in a lower detection accuracy.

The detection rate was also highly dependent on the crown category of the individual trees, that is, the relative crown position and size in comparison to adjacent trees. Fig. 3 displays the detection rates of the layer-by-layer segmentation and the watershed delineation for each category. For dominant and co-dominant trees (categories A and B, respectively), the detection rates of the two algorithms were comparable at 89.5% and 77.9% for the layer-by-layer segmentation and 86.4% and 75.2% for the watershed delineation. On the contrary, for the suppressed trees (categories C and D), the difference in performance was notable. For crown category C, the detection rate of the layer-by-layer segmentation was 42.3%, more than double the corresponding rate of 20.1% achieved by the watershed delineation. The difference was even more apparent in category D, with the respective detection rates of 15.2% and

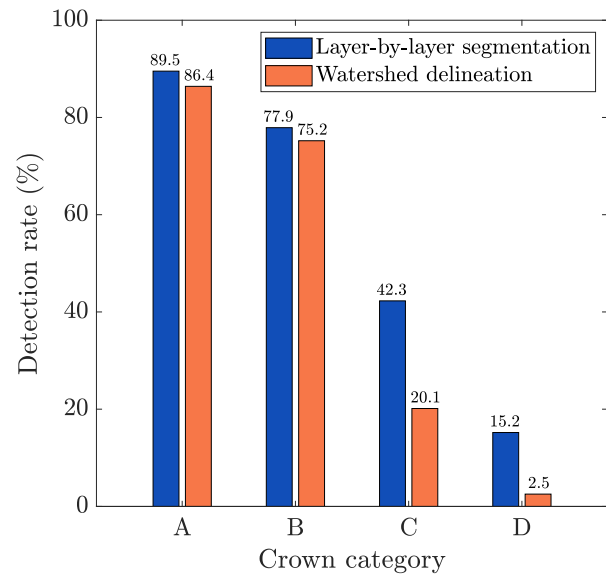


Fig. 3. Comparison of crown category-wise detection rates between the layer-by-layer segmentation and watershed delineation. Categories are dominant (A), co-dominant (B), tree alongside a dominant tree (C), and tree under a dominant tree (D).

2.5%. Since the crowns of suppressed trees are frequently partially or fully covered by adjacent dominant trees, it is usually essentially impossible to distinguish them from the CHM, which the watershed delineation uses for tree detection. On the other hand, the layer-by-layer segmentation possesses the theoretical capabilities to detect suppressed trees, since the tree detection process is based on lower layers of the point cloud. Fig. 4 illustrates an example of a situation where the watershed delineation fails to detect a suppressed tree while the layer-by-layer segmentation succeeds. Consequently, the considerable disparity in the detection accuracies of crown categories C and D between the two algorithms was expected and the performance improvement in detection of suppressed trees comprised a majority of the 23% overall increase in detected trees, as the number of trees in each crown category was approximately equal.

A relatively high detection rate for tree crown category A was expected from layer-by-layer segmentation, since the crown and stem of dominant trees are generally well defined. A failure to detect a tree in category A was typically a result of merging it to an adjacent tree, which was most common on dense plots. For the co-dominant trees (category B), we observed a tendency to merge adjacent trees of the same category

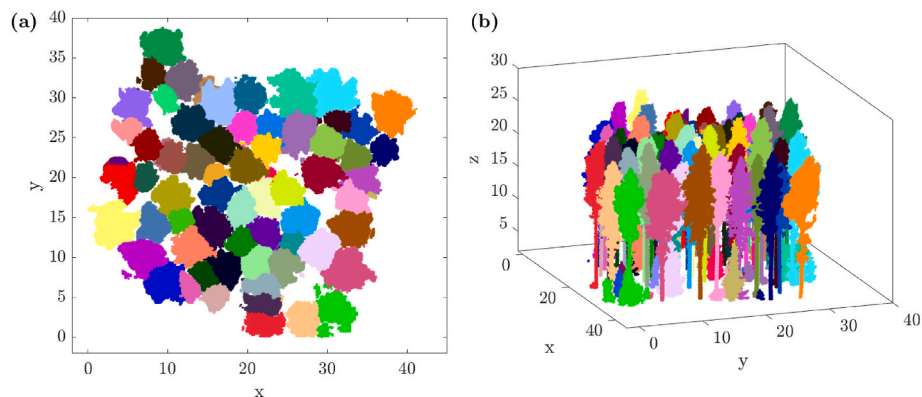


Fig. 2. Result of individual tree segmentation for one plot. Note that only segments within the area covered by the reference data are included in the figure. (a) View from the top. (b) Three-dimensional view.

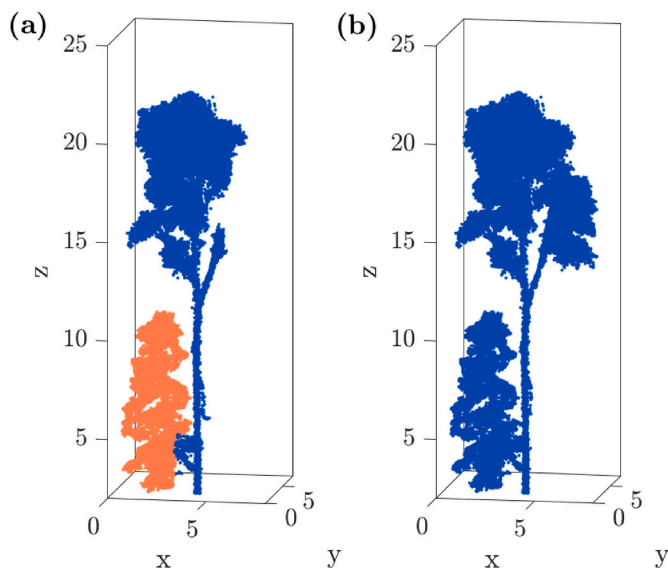


Fig. 4. Segmentation results for a small subsection that contains a dominant and a suppressed tree. Individual segments are coloured with distinct colours. (a) Segments formed with the layer-by-layer segmentation. The algorithm successfully detects both trees. (b) Segments formed with the watershed delineation. The algorithm fails to detect the suppressed tree and merges it into the dominant tree. Note how the segment contains the entire dominant tree, while a branch is missing from (a) due to being misclassified to an adjacent tree. (For interpretation of the references to colour in this figure legend, the reader is referred to the Web version of this article.)

into one if the lower sections of the trees were not adequately separated. Again, this behaviour was most common on dense plots. While the layer-by-layer segmentation achieved a considerably higher accuracy for categories C and D in comparison to the watershed delineation, the detection rates are still low in proportion to categories A and B. The most typical reasons for failing to detect a tree in category C or D were as follows:

1. The point cloud of the tree included too few points to be feasible to detect. Specifically on dense plots, the thick canopy results in fewer returns that correspond to the suppressed tree.
2. The suppressed tree was merged to an adjacent dominant tree. This was especially a problem for trees in crown category D, as they were by definition considerably closer to the dominant tree.
3. The distance between the smaller tree's top and branches of a dominant tree was so small that the gap between the two was not detected when constructing the training data. As a result, part of the adjacent tree was erroneously classified to the suppressed tree, thus the resulting segment became significantly higher than the tree it corresponded to. Consequently, the segment could not be matched to the reference data.

4.2. Results for the classification of tree species

The results of the tree species classification are reported for all trees,

Table 7

Description of the four combinations of features considered. The table contains the scanners that were used to obtain the features, the total number of features in the set, and the selected number of features for the classification.

Classification model	Feature subset	Number of features	Number of selected features
1	geometric features of scanner 1	34	9
2	all features of scanner 1	83	17
3	all features of scanner 1 and scanner 2	132	17
4	all features of all scanners	249	27

Table 8

Five best features obtained in feature selection for each classification model. The features are ordered from left to right such that the most important feature is the leftmost. The total number of chosen features after feature selection was 9, 17, 17, and 27, for the models 1 through 4, respectively. Explanations of the symbols can be found in Table 5.

Classification model	Five best features				
1	D_{only} ,	D_{last} ,	D_{10} ,	D_{first} ,	$D_{intermediate}$
2	$I_{hist, 19}^1$,	$I_{hist, 22}^1$,	I_{kurt}^1 ,	D_4 ,	D_{first}
3	$I_{hist, 19}^1$,	$I_{hist, 22}^2$,	I_{kurt}^1 ,	$I_{hist, 20}^2$,	$I_{hist, 20}^1$
4	$I_{hist, 22}^2$,	$I_{hist, 19}^1$,	I_{kurt}^1 ,	$I_{hist, 20}^2$,	R_{kurt}^1

and, in addition, separately for the four different crown categories of trees (Table 3). We compared four models in the tree species classification, which corresponded to different combinations of features (Table 7). Classification model 1 used only the geometric features of scanner 1 without including any reflectance features. In model 2, also the reflectance features of scanner 1 were included. Model 3 was a combination of all features obtained using both scanner 1 and scanner 2, and finally, the model 4 contained all features of all scanners. In all of these cases, we used the combined point cloud from which we selected points obtained with the desired scanners.

Feature selection was performed separately for each model. The models 1, 2, 3, and 4 contained 34, 83, 132, and 249 features, respectively, and the selected numbers of features for the classification based on the out-of-bag predictor were 9, 17, 17, and 27, respectively. To provide information on the most useful features for different classification models, the five best features of each model are presented in Table 8.

The performances of the models were evaluated by computing the overall accuracies and kappa coefficients, using the test set. Table 9 contains the results obtained using the selected features from models 1 and 2, and Table 10 contains the corresponding results for models 3 and 4.

Let us first compare the performances of the four classification models. We see that model 1 had the worst performance out of the four sets with an overall accuracy around 73% and kappa coefficient equal to 0.59. The overall accuracy of model 2 was 87%, so over 10 percentage points higher. The kappa coefficient using model 2 was considerably higher as well, that is, 0.80. Adding the reflectance features to the geometric features clearly improved the classification performance.

Models 1 and 2 both used only features obtained using scanner 1. By adding the data from scanner 2, we obtained model 3. We see that adding another wavelength data improved the performance even more. The overall accuracy with model 3 was approximately 90%, and the kappa coefficient was 0.85. Adding the data from the green lidar system (scanner 3) seemed to improve the performance even further since with model 4 the overall accuracy was approximately 91% and the kappa coefficient was 0.86. However, the difference is not substantial, and could be caused by the randomness of the model as well. In crown category D, the overall accuracy of model 4 was 2.5 percentage points higher than that of model 3. Scanner 3 had a high point density, and suppressed trees may therefore have been better visible in the point cloud that model 4 utilised compared with the point clouds collected

Table 9

Results of the tree species classification for models 1 and 2. Overall accuracies (OAs) and kappa coefficients are computed both for all trees in the test set and separately for trees in different crown categories.

Tree crown category	Model 1					Model 2				
	All	A	B	C	D	All	A	B	C	D
OA (%)	73.2	77.3	74.3	64.8	64.6	86.6	88.7	91.1	78.9	70.7
Kappa	0.59	0.65	0.59	0.47	0.34	0.80	0.83	0.86	0.68	0.45

Table 10

Results of the tree species classification for models 3 and 4. Overall accuracies (OAs) and kappa coefficients are computed both for all trees in the test set and separately for trees in different crown categories.

Tree crown category	Model 3					Model 4				
	All	A	B	C	D	All	A	B	C	D
OA (%)	89.9	92.0	91.9	87.3	70.7	90.8	92.7	93.7	86.6	73.2
Kappa	0.85	0.88	0.87	0.81	0.45	0.86	0.89	0.90	0.80	0.50

Table 11

Confusion matrix of the classified trees using model 4.

		Predicted labels			Total	Recall (%)
		Pine	Spruce	Deciduous		
Reference	Pine	552	14	22	588	93.9
	Spruce	16	433	24	473	91.5
	Deciduous	13	47	354	414	85.5
	Total	581	494	400	1475	
Precision (%)		95.0	87.7	88.5	OA = 90.8%, kappa = 0.86	

Table 12

Confusion matrix of the classified trees in crown category A using model 4.

		Predicted labels			Total	Recall (%)
		Pine	Spruce	Deciduous		
Reference	Pine	232	7	9	248	93.5
	Spruce	7	213	6	226	94.2
	Deciduous	2	13	114	129	88.4
	Total	241	233	129	603	
Precision (%)		96.3	91.4	88.4	OA = 92.7%, kappa = 0.89	

Table 13

Confusion matrix of the classified trees in crown category B using model 4.

		Predicted labels			Total	Recall (%)
		Pine	Spruce	Deciduous		
Reference	Pine	240	3	7	250	96.0
	Spruce	2	86	3	91	94.5
	Deciduous	6	11	148	165	89.7
	Total	248	100	158	506	
Precision (%)		96.8	86.0	93.7	OA = 93.7%, kappa = 0.90	

Table 14

Confusion matrix of the classified trees in crown category C using model 4.

		Predicted labels			Total	Recall (%)
		Pine	Spruce	Deciduous		
Reference	Pine	74	3	4	81	91.4
	Spruce	7	94	9	110	85.5
	Deciduous	4	11	78	93	83.9
	Total	85	108	91	284	
Precision (%)		87.1	87.0	85.7	OA = 86.6%, kappa = 0.80	

Table 15
Confusion matrix of the classified trees in crown category D using model 4.

		Predicted labels			Total	Recall (%)
		Pine	Spruce	Deciduous		
Reference	Pine	6	1	2	9	66.7
	Spruce	0	40	6	46	87.0
	Deciduous	1	12	14	27	51.9
	Total	7	53	22	82	
Precision (%)		85.7	75.5	63.6	OA = 73.2%, kappa = 0.50	

without scanner 3.

From Tables 9 and 10, we see that there is a difference between the classification accuracy of trees in different crown categories. Trees in categories A and B were classified more accurately than the trees in categories C and D. Furthermore, the trees in category D had significantly lower classification accuracy compared to the trees in other categories. The difference was not remarkable in classification model 1, but in other sets, the kappa coefficient of category D was considerably lower than other coefficients.

We were also interested in the difference between the accuracies of classification of different tree species. This difference was analysed using confusion matrices and class-wise precisions and recalls. The confusion matrix obtained using model 4 is presented in Table 11. We see that pines and spruces were classified with higher accuracy than deciduous trees, which is not surprising since the class of deciduous trees contained trees from several different species even though most of them were birches. Not all deciduous trees resemble one another, which is why their classification accuracy is lower than the accuracy for pines and spruces.

The confusion matrices of the trees in different crown categories using classification model 4 are presented in Tables 12–15. By comparing the confusion matrices, one can once again verify that the trees in category D were clearly the most difficult to classify. The tree species in other categories all had recall of over 83% and precision of over 85%. In category D, the highest recall was 87% for spruces, and the highest precision was 86% for pines. Again, the deciduous trees were the most difficult to classify with recall of 52% and precision of 64%, that is, a little over half of them were classified correctly. However, the total number of trees in category D was only 82, so a single misclassified tree affects the results quite considerably. The trees in categories A and B were classified with the highest accuracy, and most of the trees in our data belonged to these two categories.

4.3. Further discussion

The individual tree detection rates achieved in this study were comparable to typical detection rates obtained in individual tree segmentation studies conducted on ALS point clouds of similar forest types. Using the watershed delineation algorithm, Yu et al. (2017) obtained detection rates of 91.9%, 81.5%, 26.4%, and 7.2% for the tree crown categories A, B, C, and D, respectively. The corresponding detection rates for the layer-by-layer algorithm were 89.5%, 77.9%, 42.3%, and 15.2%, respectively. However, it is worth noting that although the point cloud data used by Yu et al. (2017) were from the same general area as ours, the mean density of the forest plots was 940 trees/ha, while the corresponding metric for our data was 1022 trees/ha. That is, our data set appears to be slightly denser on average, which generally makes accurate segmentation more difficult. Consequently, using an equivalent implementation of the watershed delineation on our data set, we obtained the detection rates of 86.4%, 75.2%, 20.1%, and 2.5% for categories A, B, C, and D, respectively. Based on the above observations, we believe that our proposed algorithm shows promise, especially for the segmentation of suppressed trees.

Out of the popular algorithms for individual tree segmentation from

ALS point clouds, layer stacking (Ayrey et al., 2017) is perhaps the most similar to the layer-by-layer segmentation. Layer stacking detects trees based on an overlap map constructed by stacking layers of clusters formed with the k-means clustering algorithm (Hartigan and Wong, 1979). Ayrey et al. (2017) obtained a detection rate of 66%–69% on dense uneven-aged conifer forests, 70% on dense mixedwood forests, and 77%–89% on even-aged conifers, including both dense and spaced forest stands. These results are similar to the performance of the layer-by-layer segmentation algorithm. However, in general it is rather difficult to compare the results of individual tree segmentation due to significant differences in test areas, point densities, and procedures used for matching segments to reference data. Furthermore, studies on individual tree segmentation frequently focus on sparser forest plots, while our data set consisted primarily of comparatively dense forests.

While detection rate of suppressed trees appeared quantitatively high in the initial tree detection, many of those trees were erroneously segmented and discarded at a later stage due to a failure to detect a gap between the crown and the branches of the overlapping trees. Providing the FkNN classifier with additional, non-distance-based, features, such as point density or reflectance, may help discern the crown from the overlapping branches during the segmentation phase more consistently. Further research is needed to determine the viability of such an approach. Furthermore, the use of more intricate machine learning methods in place of the FkNN classifier or the initial tree detection algorithm should also be considered in future work.

A notable drawback of the layer-by-layer segmentation algorithm is the required computational resources in comparison to more straightforward methods for individual tree segmentation. On a laptop with 2.6 GHz Intel i7-10750H processor and 64 GB of RAM, the layer-by-layer segmentation requires 2–5 min on average for segmenting a single forest plot, while the corresponding time for the watershed delineation is a few seconds. Performance improvements can probably be achieved, by dividing the forest plot into smaller segments during the tree detection phase, for example. However, in its current state the layer-by-layer segmentation is a rational choice only when the number of suppressed trees (categories C and D) is considerable.

In this study, we used VUX-1HA as our primary scanner, because it provided the densest and most accurate 3D point cloud. We then investigated how adding more scanners and thus wavelengths improved the species classification accuracy. Point clouds from VUX-1HA and miniVUX-3UAV were used in the three first classification models, because their infrared and near infrared wavelengths are typically used in airborne and drone-based laser scanning. In future, it would be interesting to test all possible scanner combinations and perform deeper analyses.

It is relatively difficult to compare studies on tree species classification carried out in different test areas and with different point cloud characteristics. In Yu et al. (2017), classification of pine, spruce, and birch was performed using Optech Titan airborne laser scanner. When using only geometric features, geometric features combined with intensity channel of 1550 nm, and multispectral geometric features and intensities, the classification accuracies were 76%, 82%, and 86%, respectively. We obtained the corresponding values of 73.1%, 86.6%, and 90.8%, respectively. In our study, the point density was significantly

higher than in Yu et al. (2017), but the results indicate that geometric features did not explain the species any better. It is highly probable that improved classification was due to improved intensity through higher point density, possibly due to improved segmentation methods but also due to improved features. In Axelsson et al. (2018), 179 mature solitary trees from nine species were classified using multispectral ALS Optech Titan data. The best model, whose cross-validation accuracy was 76.5%, used both geometric features and reflectance (intensity)-based features. As a comparison, when using geometric features, an accuracy of 43.0% was achieved.

Compared to our study, improved results in segmentation and species classification can probably be obtained 1) when combining leaf-off and leaf-on data, 2) by having an optimum timing for data acquisition, and 3) by combining lidar measurements with aerial images. Related to improvement of segmentation, Chen et al. (2022) compared the performance of various individual tree segmentation methods in dense deciduous forests with a fused leaf-off and leaf-on point cloud (FULD) and a regular leaf-on point cloud. In their study, layer stacking segmentation in combination with the FULD provided a 12.3% improvement in overall detection rate, with F_1 -scores ranging from 0.70 to 0.85. The density of the forest plots utilised in the study ranged from approximately 840 trees/ha to 1060 trees/ha, which is comparable to the mean density 1022.44 trees/ha of our data set. Related to improvement of species classification, dominant tree species were classified at a plot level by Kukkonen et al. (2019b). They used aerial images with multispectral laser scanner data collected using Optech Titan, achieving an overall accuracy of 88%–89%. Lindberg et al. (2021) classified deciduous and mixed species, spruces, and pines in mature and middle-aged stands. Their novel mini raster cell method achieved an overall accuracy of 75% at a plot level, whereas the overall accuracy of the area-based method was 68%. Since in our paper 93% classification overall accuracy was achieved at individual tree level for dominant and co-dominant trees, the results seem to be promising, but to find the best approaches to classify individual tree species, the authors are proposing international benchmarking studies to be conducted. The same data and the same test site but different approaches competing seems to be the only way to find out the next steps needed in the field.

5. Conclusions

In this paper, we demonstrate that the overall accuracy of tree species classification can be improved from 73% up to 91%, when moving from single-wavelength geometric features to combined geometric and multispectral reflectance features. Our tests were conducted using a multispectral close-range airborne laser scanner system in 53 forest plots with a size of 32 m × 32 m each. For dominant and co-dominant trees, classification accuracies were as high as 92.7% and 93.7%, respectively. We applied wavelengths of 532 nm, 905 nm, and 1550 nm, corresponding to Riegl VQ-840-G, miniVUX-3UAV, and VUX-1HA scanners (Riegl GmbH, Austria). In addition, a tree detection algorithm proposed by Hyyppä et al. (2020a) was developed further and extended, resulting in a new tree detection and segmentation method. The new algorithm was shown to outperform the watershed segmentation, especially when detecting and segmenting suppressed trees. Using the new algorithm, trees in the dominant and co-dominant categories were found with detection rates of 89.5% and 77.9%, respectively, whereas suppressed trees were detected with a success rate of 15.2%–42.3%. The improved detection of suppressed trees has impact on assessing forest biomass, forest growth, and regeneration ability of forests. The developed segmentation and classification methods can be applied to process dense multispectral point cloud data from airborne and drone laser scanning surveys. These in combination with information extraction for stem curve and volume (Hyyppä et al., 2022) provide means for efficient and robust field reference data for forest inventory purposes.

Author contributions

L.R. and A.H. acted as the joint first author of equal contribution. E. H. and X.Y. provided the individual tree detection algorithms based on Hyyppä et al. (2022) and Yu et al. (2011) as a starting platform for L.R. in tree detection algorithm development. X.Y. provided previously used features such as in Yu et al. (2017) for A.H. for further tree species classification development. A.H. and M.L. developed the further features. M.L. provided the ground point and DTM extraction algorithm. E. H. provided the lowest point extraction algorithm, which was used in the ground point extraction algorithm. Data processing in the article was done by A.H. and L.R., and advised with machine learning by M.L., L.M., J.T. and J.H. A.K. and H.K. developed the multispectral ALS system, made the ALS surveys, and georeferenced the ALS data. M.L. created the multispectral point clouds from three separate point clouds, corresponding to the three scanners. J.H. designed the experimental set up of the study and he provided the major funding for the work acknowledged below. Study was jointly supervised by J.H., A.K., H.K., L.M., M.L., and X.Y. Field reference was provided by V.L., V.K., and M.H. The article was mainly written by L.R., A.H., and J.H. but all authors participated in improving the manuscript.

Acknowledgements

We gratefully acknowledge for financial support the Academy of Finland projects ‘Forest-Human-Machine Interplay’ (Academy decision 337656), ‘Capturing structural and functional diversity of trees and tree communities for supporting sustainable use of forests’ (348644), ‘Feasibility of Inside-Canopy UAV Laser Scanning for Automated Tree Quality Surveying’ (334002), ‘Mapping of forest health, species and forest fire risks using Novel ICT Data and Approaches’ (344755, Chistera Eranet), ‘Understanding Wood Density Variation Within and Between Trees Using Multispectral Point Cloud Technologies and X-ray microdensitometry’ (331708), ‘Estimating Forest Resources and Quality-related Attributes Using Automated Methods and Technologies’ (334829), and the Ministry of Agriculture and Forestry project ‘Future forest information system at individual tree level’ (VN/3482/2021). Academy of Finland Research infrastructure (346382) was applied for the study.

References

- Axelsson, A., Lindberg, E., Olsson, H., 2018. Exploring multispectral ALS data for tree species classification. *Rem. Sens.* 10, 183. <https://doi.org/10.3390/rs10020183>.
- Ayrey, E., Fraver, S., Kershaw Jr., J.A., Kenefic, L.S., Hayes, D., Weiskittel, A.R., Roth, B. E., 2017. Layer stacking: a novel algorithm for individual forest tree segmentation from LIDAR point clouds. *Can. J. Rem. Sens.* 43, 16–27. <https://doi.org/10.1080/07038992.2017.1252907>.
- Brandtberg, T., 2007. Classifying individual tree species under leaf-off and leaf-on conditions using airborne lidar. *ISPRS J. Photogrammetry Remote Sens.* 61, 325–340. <https://doi.org/10.1016/j.isprs.2006.10.006>.
- Brandtberg, T., Warner, T.A., Landenberger, R.E., McGraw, J.B., 2003. Detection and analysis of individual leaf-off tree crowns in small footprint, high sampling density lidar data from the eastern deciduous forest in North America. *Rem. Sens. Environ.* 85, 290–303. [https://doi.org/10.1016/S0034-4257\(03\)00008-7](https://doi.org/10.1016/S0034-4257(03)00008-7).
- Breiman, L., 2001. Random forest. *Mach. Learn.* 45, 5–32. <https://doi.org/10.1023/A:1010933404324>.
- Campbell, R.B., Bartos, D.L., 2001. Aspen ecosystems: objectives for sustaining biodiversity. In: Shepperd, W.D., Binkley, D., Bartos, D.L., Stohlgren, T.J., Eskew, L. G. (Eds.), *Compilers. Sustaining Aspen in Western Landscapes: Symposium Proceedings. Proceedings RMRS-P-18. USDA Forest Service, Rocky Mountain Research Station, Fort Collins, Colorado, USA*, pp. 299–307.
- Campos, M.B., Litkey, P., Wang, Y., Chen, Y., Hyyti, H., Hyyppä, J., Puttonen, E., 2021. A long-term terrestrial laser scanning measurement station to continuously monitor structural and phenological dynamics of boreal forest canopy. *Front. Plant Sci.* 11, 606752 <https://doi.org/10.3389/fpls.2020.606752>.
- Chen, Q., 2007. Airborne lidar data processing and information extraction. *Photogramm. Eng. Rem. Sens.* 73, 109–112.
- Chen, Q., Gao, T., Zhu, J., Wu, F., Li, X., Lu, D., Yu, F., 2022. Individual tree segmentation and tree height estimation using leaf-off and leaf-on UAV-LiDAR data in dense deciduous forests. *Rem. Sens.* 14 <https://doi.org/10.3390/rs14122787>.

- Dalponte, M., Coomes, D.A., 2016. Tree-centric mapping of forest carbon density from airborne laser scanning and hyperspectral data. *Methods Ecol. Evol.* 7, 1236–1245. <https://doi.org/10.1111/2041-210X.12575>.
- Edson, C., Wing, M.G., 2011. Airborne light detection and ranging (LiDAR) for individual tree stem location, height, and biomass measurements. *Rem. Sens.* 3, 2494–2528. <https://doi.org/10.3390/rs3112494>.
- Ester, M., Kriegel, H.P., Sander, J., Xu, X., et al., 1996. A density-based algorithm for discovering clusters in large spatial databases with noise. In: *Proceedings of 2nd International Conference on Knowledge Discovery and Data Mining*, pp. 226–231 (KDD-96).
- Fassnacht, F.E., Latifi, H., Stereńczak, K., Modzelewska, A., Lefsky, M., Waser, L.T., Straub, C., Ghosh, A., 2016. Review of studies on tree species classification from remotely sensed data. *Rem. Sens. Environ.* 186, 64–87.
- Gamfeldt, L., Snäll, T., Bagchi, R., Jonsson, M., Gustafsson, L., Kjellander, P., Ruiz-Jaen, M.C., Fröberg, M., Stendahl, J., Philipson, C.D., et al., 2013. Higher levels of multiple ecosystem services are found in forests with more tree species. *Nat. Commun.* 4, 1340.
- Harris, C.R., Millman, K.J., van der Walt, S.J., Gommers, R., Virtanen, P., Cournapeau, D., Wieser, E., Taylor, J., Berg, S., Smith, N.J., Kern, R., Picus, M., Hoyer, S., van Kerkwijk, M.H., Brett, M., Haldane, A., del Río, J.F., Wiebe, M., Peterson, P., Gérard-Marchant, P., Sheppard, K., Reddy, T., Weckesser, W., Abbasi, H., Gohlke, C., Oliphant, T.E., 2020. Array programming with NumPy. *Nature* 585, 357–362. <https://doi.org/10.1038/s41586-020-2649-2>.
- Hartigan, J.A., Wong, M.A., 1979. Algorithm as 136: a k-means clustering algorithm. *J. Royal Statist. Soc. Series C (Applied Statistics)* 28, 100–108. <https://doi.org/10.2307/2346830>.
- Holmgren, J., Persson, Å., 2004. Identifying species of individual trees using airborne laser scanner. *Rem. Sens. Environ.* 90, 415–423. [https://doi.org/10.1016/S0034-4257\(03\)00140-8](https://doi.org/10.1016/S0034-4257(03)00140-8).
- Huuskonen, S., Domisch, T., Finér, L., Hantula, J., Hynynen, J., Matala, J., Miina, J., Neuvonen, S., Nevalainen, S., Niemistö, P., Nikula, A., Piri, T., Siitonen, J., Smolander, A., Tonteri, T., Uotila, K., Viiri, H., 2021. What is the potential for replacing monocultures with mixed-species stands to enhance ecosystem services in boreal forests in fennoscandia? *For. Ecol. Manag.* 479, 118558. <https://doi.org/10.1016/j.foreco.2020.118558>. <https://www.sciencedirect.com/science/article/pii/S037811272031327X>.
- Hyypä, E., Hyyppä, J., Hakala, T., Kukko, A., Wulder, M.A., White, J.C., Pyörälä, J., Yu, X., Wang, Y., Virtanen, J.P., Pohjavirta, O., Liang, X., Holopainen, M., Kaartinen, H., 2020a. Under-canopy UAV laser scanning for accurate forest field measurements. *ISPRS J. Photogrammetry Remote Sens.* 164, 41–60. <https://doi.org/10.1016/j.isprsjprs.2020.03.021>.
- Hyypä, E., Kukko, A., Kaartinen, H., Yu, X., Muhojoki, J., Hakala, T., Hyyppä, J., 2022. Direct and automatic measurements of stem curve and volume using a high-resolution airborne laser scanning system. *Sci. Rem. Sens.* 5, 100050. <https://doi.org/10.1016/j.srs.2022.100050>.
- Hyypä, E., Kukko, A., Kajaluoto, R., White, J.C., Wulder, M.A., Pyörälä, J., Liang, X., Yu, X., Wang, Y., Kaartinen, H., Virtanen, J.P., Hyyppä, J., 2020b. Accurate derivation of stem curve and volume using backpack mobile laser scanning. *ISPRS J. Photogrammetry Remote Sens.* 161, 246–262. <https://doi.org/10.1016/j.isprsjprs.2020.01.018>.
- Keller, J.M., Gray, M.R., Givens, J.A., 1985. A fuzzy k-nearest neighbor algorithm. *IEEE Transactions on Systems, Man, and Cybernetics SMC-15* 580–585. <https://doi.org/10.1109/TSMC.1985.6313426>.
- Koenig, K., Höfle, B., 2016. Full-waveform airborne laser scanning in vegetation studies—a review of point cloud and waveform features for tree species classification. *Forests* 7, 198. <https://doi.org/10.3390/f7090198>.
- Korpela, I., Ørka, H.O., Maltamo, M., Tokola, T., Hyyppä, J., et al., 2010. Tree species classification using airborne LiDAR—effects of stand and tree parameters, downsizing of training set, intensity normalization, and sensor type. *Silva Fenn.* 44, 319–339. <https://doi.org/10.14214/sf.156>.
- Kukkonen, M., Maltamo, M., Korhonen, L., Packalen, P., 2019a. Comparison of multispectral airborne laser scanning and stereo matching of aerial images as a single sensor solution to forest inventories by tree species. *Rem. Sens. Environ.* 231, 111208. <https://doi.org/10.1016/j.rse.2019.05.027>.
- Kukkonen, M., Maltamo, M., Korhonen, L., Packalen, P., 2019b. Multispectral airborne LiDAR data in the prediction of boreal tree species composition. *IEEE Trans. Geosci. Rem. Sens.* 57, 3462–3471. <https://doi.org/10.1109/TGRS.2018.2885057>.
- Lehtomäki, M., Jaakkola, A., Hyyppä, J., Lampinen, J., Kaartinen, H., Kukko, A., Puttonen, E., Hyyppä, H., 2016. Object classification and recognition from mobile laser scanning point clouds in a road environment. *IEEE Trans. Geosci. Rem. Sens.* 54, 1226–1239. <https://doi.org/10.1109/TGRS.2015.2476502>.
- Liang, X., Hyyppä, J., Matikainen, L., 2007. Deciduous-coniferous tree classification using difference between first and last pulse laser signatures. *Int. Arch. Photogram. Rem. Sens. Spatial Inf. Sci.* 36 (3/W52).
- Lindberg, E., Holmgren, J., Olsson, H., 2021. Classification of tree species classes in a hemi-boreal forest from multispectral airborne laser scanning data using a mini raster cell method. *Int. J. Appl. Earth Obs. Geoinf.* 100, 102334. <https://doi.org/10.1016/j.jag.2021.102334>.
- Mathworks, 2022. Predictor importance estimates by permutation of out-of-bag predictor observations for random forest of classification trees. URL: <https://se.mathworks.com/help/stats/classificationbaggedensemble.oobpermutedpredictorimportance.html>, 2022-12-09.
- McKinney, Wes, 2010. Data structures for statistical computing in Python. In: *Proceedings of the 9th Python in Science Conference*, pp. 56–61. <https://doi.org/10.25080/Majora-92b1922-00a>.
- Michałowska, M., Rapiński, J., 2021. A review of tree species classification based on airborne lidar data and applied classifiers. *Rem. Sens.* 13, 353.
- Moffiet, T., Mengersen, K., Witte, C., King, R., Denham, R., 2005. Airborne laser scanning: exploratory data analysis indicates potential variables for classification of individual trees or forest stands according to species. *ISPRS J. Photogrammetry Remote Sens.* 59, 289–309. <https://doi.org/10.1016/j.isprsjprs.2005.05.002>.
- Ørka, H.O., Næsset, E., Bollandsås, O.M., 2009. Classifying species of individual trees by intensity and structure features derived from airborne laser scanner data. *Rem. Sens. Environ.* 113, 1163–1174. <https://doi.org/10.1016/j.rse.2009.02.002>.
- Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., Duchesnay, E., 2011. Scikit-learn: machine learning in Python. *J. Mach. Learn. Res.* 12, 2825–2830.
- Persson, A., Holmgren, J., Söderman, U., 2006. Identification of tree species of individual trees by combining very high resolution laser data with multi-spectral images. In: *Proceedings of the International Workshop on 3D Remote Sensing in Forestry*, pp. 102–107.
- Roussel, J.R., Auty, D., 2022. Airborne LiDAR Data Manipulation and Visualization for Forestry Applications. R package version 4.0.1.
- Roussel, J.R., Auty, D., Coops, N.C., Tompalski, P., Goodbody, T.R., Meador, A.S., Bourdon, J.F., de Boissieu, F., Achim, A., 2020. lidR: an R package for analysis of airborne laser scanning (ALS) data. *Rem. Sens. Environ.* 251, 112061. <https://doi.org/10.1016/j.rse.2020.112061>.
- Seidl, R., Thom, D., Kautz, M., Martin-Benito, D., Peltoniemi, M., Vacchiano, G., Wild, J., Ascoli, D., Petr, M., Honkaniemi, J., et al., 2017. Forest disturbances under climate change. *Nat. Clim. Change* 7, 395–402.
- Silva, C.A., Hudak, A.T., Vierling, L.A., Loudermilk, E.L., O'Brien, J.J., Hiers, J.K., Jack, S.B., Gonzalez-Benecke, C., Lee, H., Falkowski, M.J., Khosravipour, A., 2016. Imputation of individual longleaf pine (*Pinus palustris* mill.) tree attributes from field and LiDAR data. *Can. J. Rem. Sens.* 42, 554–573. <https://doi.org/10.1080/07038992.2016.1196582>.
- The pandas development team, 2020. pandas-dev/pandas: Pandas. <https://doi.org/10.5281/zenodo.3509134>.
- Van Rossum, G., Drake, F.L., 2009. *Python 3 Reference Manual*. CreateSpace, Scotts Valley, CA.
- Wichmann, V., Bremer, M., Lindenberger, J., Rutzinger, M., Georges, C., Petrini-Monteferrri, F., 2015. Evaluating the potential of multispectral airborne lidar for topographic mapping and land cover classification. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences II-3/W5* 113–119.
- Yang, J., Kang, Z., Cheng, S., Yang, Z., Akwensi, P.H., 2020. An individual tree segmentation method based on watershed algorithm and three-dimensional spatial distribution analysis from airborne LiDAR point clouds. *IEEE J. Sel. Top. Appl. Earth Obs. Rem. Sens.* 13, 1055–1067. <https://doi.org/10.1109/JSTARS.2020.2979369>.
- Yu, X., Hyyppä, J., Kukko, A., Maltamo, M., Kaartinen, H., 2006. Change detection techniques for canopy height measurements using airborne laser scanner data. *Photogramm. Eng. Rem. Sens.* 72, 1339–1348. <https://doi.org/10.14358/PERS.72.12.1339>.
- Yu, X., Hyyppä, J., Vastaranta, M., Holopainen, M., Viitala, R., 2011. Predicting individual tree attributes from airborne laser point clouds based on the random forests technique. *ISPRS J. Photogrammetry Remote Sens.* 66, 28–37. <https://doi.org/10.1016/j.isprsjprs.2010.08.003>.
- Yu, X., Hyyppä, J., Litkey, P., Kaartinen, H., Vastaranta, M., Holopainen, M., 2017. Single-sensor solution to tree species classification using multispectral airborne laser scanning. *Rem. Sens.* 9, 108. <https://doi.org/10.3390/rs9020108>.