OXFORD

Institute of Chartered Foresters

Forestry: An International Journal of Forest Research, 2024, 1–15

https://doi.org/10.1093/forestry/cpae044

Software Article

Testing *treecbh* in Central European forests: an R package for crown base height detection using high-resolution aerial laser-scanned data

Gergő Diószegi[1,*], Vanda Éva Molnár[2,3], Loránd Attila Nagy[1], Péter Enyedi[4], Péter Török[3,5], Szilárd Szabó[1]

[1]Department of Physical Geography and Geoinformatics, University of Debrecen, Egyetem tér 1, 4032 Debrecen, Hungary
[2]HUN-REN- UD Anthropocene Ecology Research Group, Egyetem tér 1, 4032 Debrecen, Hungary
[3]Department of Ecology, University of Debrecen, Egyetem tér 1, 4032 Debrecen, Hungary
[4]Envirosense Hungary Ltd., Péchy M. u. 46, 4032 Debrecen, Hungary
[5]HUN-REN- UD Functional and Restoration Ecology Research Group, Egyetem tér 1, 4032 Debrecen, Hungary
[*]Corresponding author. Department of Physical Geography and Geoinformatics, University of Debrecen, Egyetem tér 1, Debrecen, Hungary.
E-mail: dioszegi.gergo@science.unideb.hu

Abstract

Accurate information regarding tree canopy characteristics is crucial for forest management, but it is often difficult to assess. This study presents an innovative framework designed for crown base height (CBH) detection using high-resolution laser-scanned data, with a specific focus on individual trees within forests. The framework comprises three key steps: (i) segmenting the input tree point cloud to identify the tree trunk and its branches using the *treesio* software; (ii) applying vertical cross-sectional K-means clustering to cluster the identified tree and to define the elevation threshold for removing low-lying understory vegetation; (iii) employing a novel 2D kernel method for detecting CBH after eliminating low-lying understory vegetation. The 2D kernel method, developed for broadleaf forests using leaf-off airborne laser scanning (ALS) data, underpins the *treecbh* tool. This tool features a visual CBH adjustment component that shows a 2D profile plot of the tree point cloud, and suggests a CBH value for user approval or adjustment. To evaluate accuracy, *in situ* measured CBH data from five forest plots in Germany and Hungary with varied species compositions were used. ALS data were collected during leaf-off conditions for the two Hungarian plots and during leaf-on conditions for the three German plots. Leaf-off terrestrial laser-scanned data from individual trees were also used in the accuracy assessment. A sensitivity analysis using random point decimation was conducted on the terrestrial laser-scanned data to assess *treecbh*'s sensitivity to point density. The initial results exhibited matching rates of 45% and 60% for leaf-off ALS plots, which significantly improved to 71% and 77%, respectively, when using the visual CBH adjustment feature of the tool. The leaf-on ALS results demonstrated matching rates between 24% and 33%, whereas the CBHs of individual terrestrial laser-scanned trees could be detected with 93% accuracy in visual mode. It was observed that *treecbh* operates effectively when the input ALS data have a minimum point density of 20 pts/m^2, with its optimal performance achieved at 110 pts/m^2. These findings indicated *treecbh*'s sensitivity to ALS data quality, scanning season (leaf-on and leaf-off), and point density. This sensitivity can be effectively mitigated in the case of leaf-off ALS data by utilizing the visual CBH adjustment feature of the tool.

Keywords: crown base height detection; tree isolation; broadleaf forest; individual tree level; ALS; R

Introduction

Crown base height (CBH) is defined as the vertical distance from the ground surface to the lowest live branch (Hermosilla et al. 2014; Luo et al. 2018; Maguya et al. 2015; Næsset and Økland 2002; Popescu and Zhao 2008; Stefanidou et al. 2020; Xu et al. 2013). It is a crucial structural parameter used to estimate the crown volume of individual trees and forest stands (Korhonen et al. 2013; Terryn et al. 2023), evaluate crown conditions (Bianchi et al. 2020) and forest health (Zarnoch et al. 2004), and determine various canopy fuel parameters (Andersen et al. 2005; Engelstad et al. 2019; Erdody and Moskal 2010; González-Ferreiro et al. 2013). In this context, accurate CBH estimations are vital for forest fire prediction simulations (Finney 1998, 2006; Kelly et al. 2018; Riaño et al. 2003).

However, obtaining accurate CBH measurements remains a challenge. Traditional field surveys yield reliable data but are time consuming, labor intensive, and expensive (Dean et al. 2009; Luo et al. 2018; Stefanidou et al. 2020). Remote sensing technology using aerial laser scanning (ALS) could be a more cost-effective alternative. ALS has been proven to be effective in providing CBH measures for individual trees and forest plots (Andersen et al. 2005; Erdody and Moskal 2010; Luo et al. 2018; Maguya et al. 2015; Stefanidou et al. 2020).

Previous studies have mainly focused on CBH estimations at the plot level (Andersen et al. 2005; Jakubowksi et al. 2013; Maguya et al. 2015), while the individual tree-level approach has received less attention (Luo et al. 2018; Popescu and Zhao 2008; Vauhkonen 2010). Estimations at the individual tree level can provide more precise information on forest structure (Luo et al.

Handling editor: Dr. Fabian Fassnacht
Received 25 January 2024. **Revised** 5 August 2024. **Accepted** 12 August 2024

2018) and become increasingly relevant in the context of precision forestry and also to describe local habitat structures. Most of the previously mentioned studies focusing on CBH estimation from laser scanning data have concentrated on coniferous forests (Botequim et al. 2019; Erdody and Moskal 2010; Kelly et al. 2018; Luo et al. 2018; Næsset and Økland 2002; Popescu and Zhao 2008; Vauhkonen 2010), whereas corresponding approaches for broadleaf trees are very sparse. More importantly, none of the aforementioned studies have provided practical software tools for CBH detection.

Previous studies of ALS-derived CBH estimation methods fall into two main categories: regression analysis (Botequim et al. 2019; Erdody and Moskal 2010; Hermosilla et al. 2014; Kelly et al. 2018; Næsset and Økland 2002) and direct CBH estimation (Dean et al. 2009; Luo et al. 2018; Popescu and Zhao 2008; Vauhkonen 2010; Vauhkonen et al. 2012). Regression-based models rely on extensive field data and are limited by sample quantity and quality (Stefanidou et al. 2020), making them site and species specific (Engelstad et al. 2019). Direct methods do not rely on field measurements for calibrating a model but rather determine CBH directly based on point cloud characteristics (Sumnall et al. 2017). However, their effectiveness depends on the pulse penetration capability, which is influenced by sensor characteristics, canopy cover, slope conditions, and data acquisition settings (Hsu et al. 2015; Sumnall et al. 2016).

This study introduces a newly developed direct CBH method, provided in the form of an open-source R package tailored for high-resolution ALS data captured within broadleaf forests. The tool named *treecbh* provides a workflow that includes three steps: tree isolation, understory removal, and CBH detection. Our specific objectives while developing the tool were (i) to provide an automated, adaptable, and easily optimizable tool for detecting CBH using point cloud data of already segmented individual trees; (ii) to validate the effectiveness of *treecbh* in detecting CBH using leaf-off and leaf-on ALS data against field-measured data obtained from central European forests with diverse compositions; and (iii) to determine the optimal point density necessary to accurately represent the structure of the input tree point cloud, particularly focusing on the tree trunk and lower branches.

The *treecbh* tool

Operating framework of *treecbh*

The *treecbh* framework operates based on a three-step concept:

1. Tree trunk isolation: In this initial phase, the framework focuses on isolating the tree trunk with its first leaved branches of the dominant tree within the input point cloud. The input point cloud is in this study obtained through manual segmentation, as detailed in Section Validation of *treecbh*: materials and methods.
2. Vertical cross-sectional K-means clustering: After tree isolation, this stage addresses the separation of understory elements from the tree trunk and its lower branches.
3. 2D kernel density method for CBH detection: Finally, the framework employs a 2D kernel density method specifically designed for detecting CBH based on the extracted tree stem and its lower branches.

Tree trunk isolation

Assuming the presence of smaller nondominant trees and understory vegetation in Central European forest stands, a point cloud–based segmentation approach was implemented to precisely identify and extract the trunk of the dominant tree along with its first leaved branches (Fig. 1a). This methodology requires a point cloud of a segmented individual tree as input and leverages the *treeiso* software, which employs two successive cut-pursuit clusterings (Xi and Hopkinson 2022). Conceptually, the point cloud was treated as a graph, where each point represents a node and the connections between points were denoted as edges. Using the cut-pursuit algorithm, the point-cloud graph first undergoes partitioning into clusters. The boundaries between these clusters are determined by minimizing the total variation within the initial cloud (Landrieu and Obozinski 2016), which corresponded to the input point cloud for *treecbh*. Subsequently, a secondary cut-pursuit clustering function was employed to merge small 3D clusters into larger ones, thereby improving the coherence of segmentation (Xi and Hopkinson 2022). The resulting point cloud is an isolated tree cloud (Fig. 1a, Tree trunk with first leaved branches), which encompasses the trunk along with its lower branches of the dominant tree isolated from the input point cloud (Fig. 1a, Input point cloud). It only contains points positioned above 0.2 m (Fig. 1a). The shape of the isolated tree point cloud can vary depending on the geometry of the input point cloud, which is influenced by the forest structure. Although it is rare, the resulting point cloud might occasionally include more than one isolated tree trunk.

Vertical cross-sectional K-means clustering

In the second step, a vertical cross-sectional cuboid 5 m wide is applied to the *xy* attributes of the isolated tree cloud. This 5-m width adequately encompasses any tree cloud in the *y* dimension, capturing points in both the *x* and *z* dimensions. This operation creates a transect through the tree cloud, defining a specific region of interest within the *xy* plane of the point cloud. This allows for the extraction of a vertical cross-section from the tree cloud, resulting in a 2D profile that displays the *xz* attributes of the points within the specified region of interest (Fig. 1b).

Next, the *minH* parameter is introduced to serve as a minimal height threshold, accounting for the probable presence of understory vegetation. Understory vegetation is represented by a higher point count than that of the tree trunk, resulting in a higher peak in the point distribution of the 2D vertical cross-section. We assume that understory vegetation is present below the 10th quantile in the distribution of the input point cloud and its 2D vertical cross-section (xz). The computation is given by Equation 1:

$$minH_i = \text{quantile10}(height_i)^{\alpha} \cdot 2 \quad (1)$$

where *i* signifies the *i*-th tree cloud and quantile10 represents the function to capture the sample quantile at the 10th percentile of the height (z) of the *i*-th tree cloud ($height_i$). The scaling parameter α facilitates a user-defined height threshold related to the height of the tree cloud, under which understory vegetation is assumed and removed internally (Fig. 1b). α is denoted by the *min_H_scale* parameter in *treecbh*, and has a default value of 0.13, which equates to 13% of the tree height.

Afterwards, the height attribute (z) of the 2D data undergoes scaling and centering processes, as required for the K-means clustering methodology. The estimation of the number of clusters defining the *k* parameter is accomplished using the prediction strength of the clustering technique (Tibshirani and Walther 2005) (Fig. 1b). Finally, the cluster characterized by the lowest height value is identified and extracted from the processed data (Fig. 1b). This extracted dataset then serves as an input for the subsequent 2D kernel density analysis.

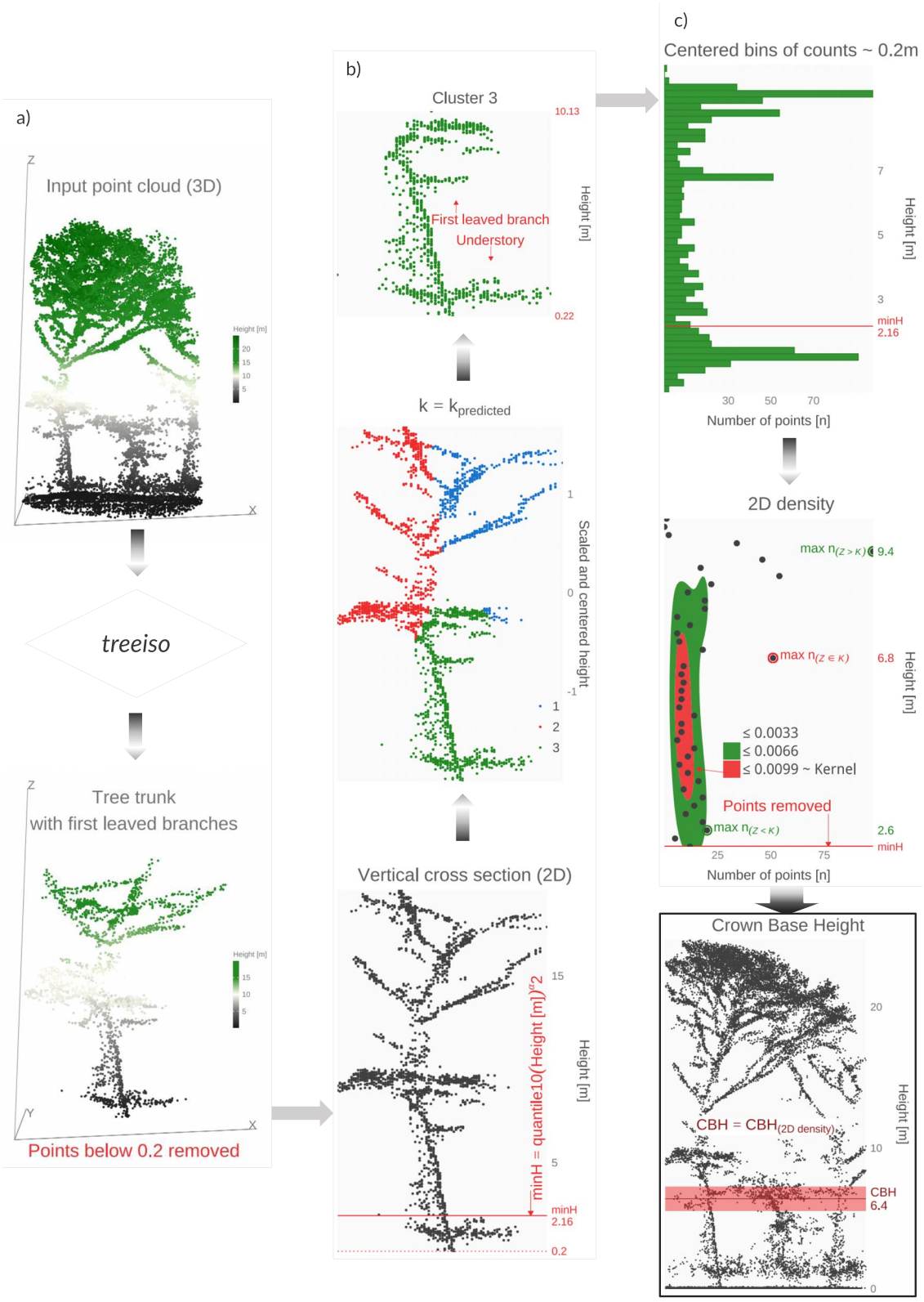

Figure 1. Framework of *treecbh*: tree isolation (a), vertical cross-sectional K-means clustering (b), and the 2D kernel method (c). The detected crown base height is illustrated in the graph at the bottom right.

Crown base height detector, the 2D kernel density method

The 2D kernel method analyzes a histogram of points from the cluster identified in the second step. In *treecbh*, the *branch_WIDTH* parameter defines the bin width of the histogram (default value of 0.2 m) with which the input points are binned (Fig. 1c, Centered bins of counts ~0.2 m). Bins with heights equal to or below the *minH* threshold (Equation 1) are systematically removed. The filtered data are then used for 2D kernel density estimation. The goal of this analysis is to find the region with the highest

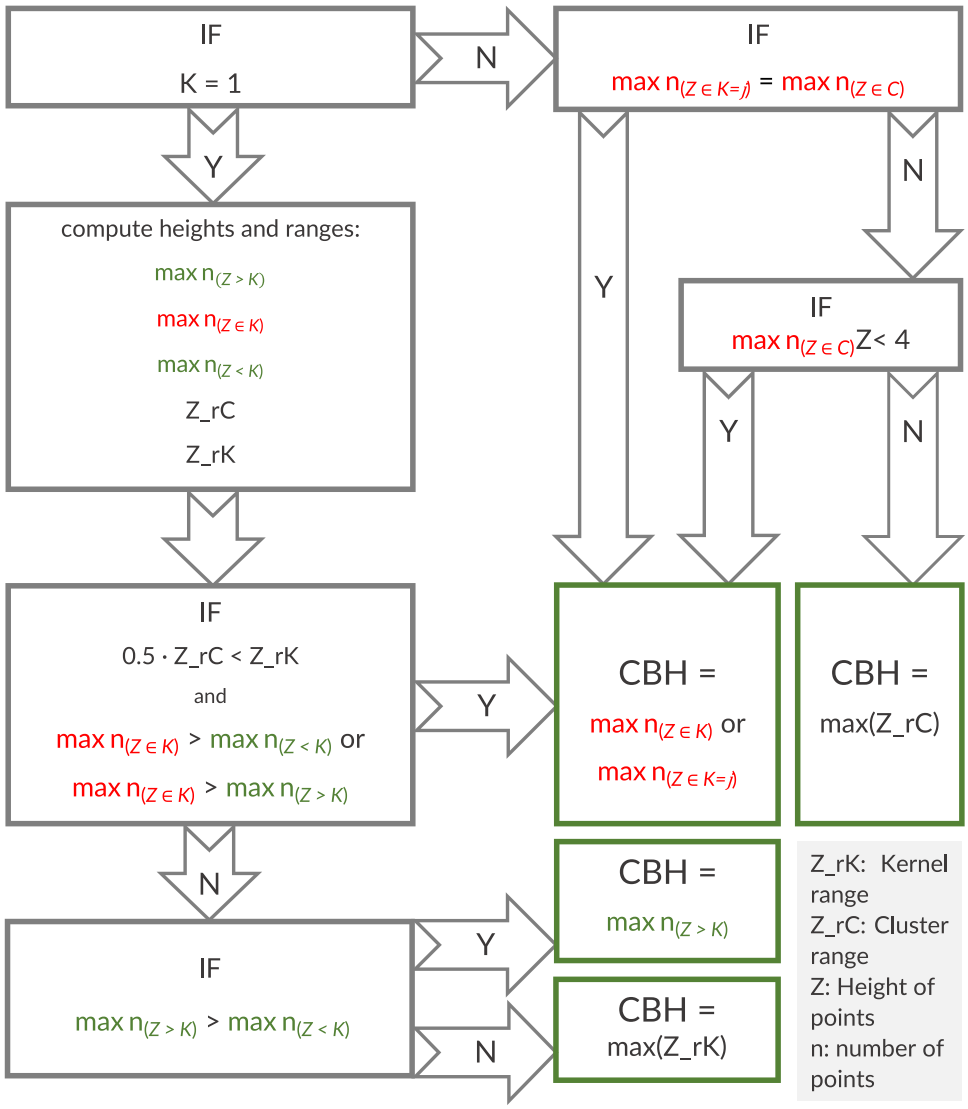

Figure 2. The workflow of the final stage of CBH identification, the decision scheme (Y: yes, N: no; K = 1 means one kernel; Z_rK and Z_rC contain two values; Z represents a vector of heights).

bin density, referred to as the "Kernel" (Fig. 1c, 2D density). The 2D kernel method applies three kernel density classes: ≤0.0033, ≤0.0066, and ≤0.0099 on the binned points (Fig. 1c, 2D density). The analysis involves three steps: determining the number of kernels (in cases of more than one isolated tree trunk), defining the range of these kernels, and establishing the maximum count of points vertically within each kernel range. For multiple kernels, the one with the highest point count within its range is selected for further analysis. Additionally, the maximum count of points both below and above the identified kernel is determined (Fig. 1c, 2D density), along with the range of the cluster (i.e. the minimum and maximum heights of the input data). This results in five candidate heights: the heights with the maximum counts below the kernel ($\mathbf{max}\ n_{(z<K)}$), within the kernel range ($\mathbf{max}\ n_{(z\in K)}$), and above the kernel ($\mathbf{max}\ n_{(z>K)}$) (Fig. 1c, 2D density), as well as the cluster height (max(Z_rC)) and kernel height (max(Z_rK)) (Fig. 2).

In the final stage of CBH identification, one of these candidate heights is selected as the CBH. This decision-making process starts with the number of kernels identified through automated 2D kernel density analysis and follows a systematic if-yes-no strategy, as illustrated in the workflow diagram (Fig. 2).

The *treecbh* tool implemented in R

The *treecbh* tool is an open-source R package designed for seamless integration within the *lidR* package. This package also allows R users to run individual tree isolation using the *treeiso* software, whose plugin is specifically available within the CloudCompare (CC) software (Girardeu-Monteau 2023). The automation of *treeiso* was facilitated through CC. In this context, *treecbh* triggers the *treeiso* plugin, allowing CC to execute the successive cut-pursuit clustering steps as part of its functionality. Thus, installation of the CC is required to run the *treecbh* package in R.

Parameters of *treecbh*

Throughout this study, two parameters, *min_H_scale* and *branch_WIDTH*, were set consistently with fixed default values and remained unchanged. However, *treecbh* incorporates two additional parameters: *cbh_ONLY* and *kM*. An in-depth overview detailing the *treecbh* parameter annotations, their implications, and the corresponding thresholds is presented in Table 1.

By setting the parameter *cbh_ONLY*, users can specify whether *treecbh* executes the complete 3D tree isolation followed by CBH detection, performs tree isolation only, or focuses solely on CBH

Table 1. Parameter characteristics of *treecbh*

Name	Default value	Implication	Range or values
*min_H_scale***	0.13	Minimum height scaler (m), controlling understory removal	[0.13–0.25]
*branch_WIDTH***	0.2	Assumed CBH branch width (m), controlling bin width for counting points	[0.01–0.5]
cbh_ONLY	1	Options for executing: 1 ~ *treeiso* and cbh, 2 ~ only *treeiso*, 3 ~ only cbh detection	[1,2,3]
*kM**	TRUE	K-means clustering (TRUE) or visual CBH adjustment (FALSE)	[–]

*Parameter that triggers visual CBH adjustment. **Parameters computed using their default values.

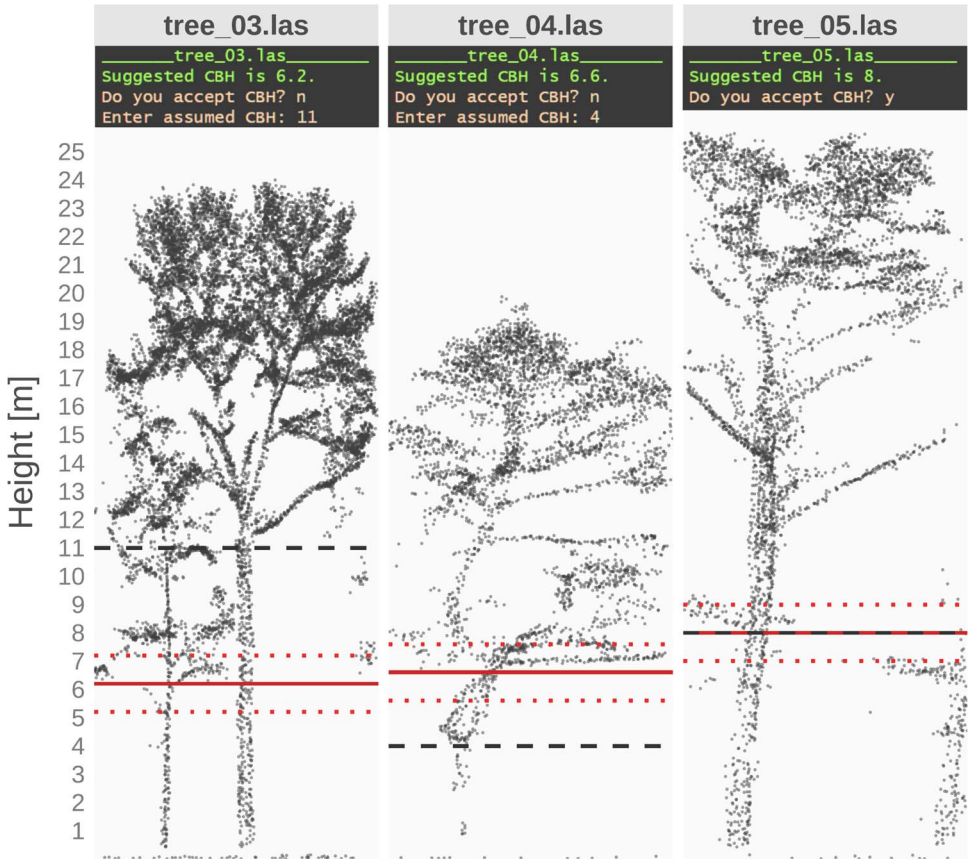

Figure 3. Example of visual CBH adjustment process triggered by kM parameter deactivation. The solid line indicates the suggested CBH by treecbh, while the dotted lines represent the suggested CBH with a range of ±1 m. The dashed line indicates the user-entered assumed CBH (not shown by the tool). R Consoles display interactive processes with user inputs.

detection. This flexibility is particularly useful when the tool is used in the visual CBH adjustment mode because it allows users to disable 3D tree segmentation (i.e. *treeiso*). When the *kM* parameter is deactivated by setting *kM* = FALSE (*treecbh* skips K-means clustering and employs only the 2D kernel technique), the interactive CBH adjustment process is initiated. The CBH detector plots the 2D profile of the input point cloud and proposes a CBH in the R Console awaiting user approval. Users can either accept or reject the proposed value while inspecting a 2D plot (Fig. 3). If the proposed value is rejected, *treecbh* prompts the user to suggest a CBH value. It then adjusts the CBH detection to the proposed height and searches for CBH within a 0.5-m distance in both the negative and positive directions of the user-defined CBH value.

Parameters of *treeiso*

If the parameter *cbh_ONLY* is set to 1 or 2 (as outlined in Table 1), it allows the customization of nine additional parameters within treeiso (Xi and Hopkinson 2022). It is important to note that the default values for these parameters remained unchanged in the context of this study. For a comprehensive understanding of the annotations, implications, and thresholds associated with the treeiso parameters detailed in Table 2, please refer to Xi and Hopkinson (2022).

R-using *treecbh*

In the context of prioritizing CBH detection, *treecbh* relies on the user-defined *kM* parameter (*kM* = FALSE) that triggers the interactive CBH adjustment process. Notably, the tool defaults to automated full tree isolation, completing the secondary cut-pursuit clustering of *treeiso* by linking cloud segments to isolate trees (as described in Section Tree trunk isolation). This default functionality accommodates users interested in isolating trees from terrestrial laser scanning data and offers flexibility in adjusting the *treeiso* parameters (Xi and Hopkinson 2022) to suit their specific needs.

Validation of *treecbh*: materials and methods

ALS forests and tree segments

Validation of the suggested workflow was conducted in three study areas: the Hardtwald forest in Karlsruhe, the Bretten

Table 2. Parameter characteristics of *treeiso*

Name	Default value	Implication	Range or values
K1	10	Number of nearest neighbors, controlling unit size of 3D cluster	[3–50]
K2	20		[5–50]
L1	1	Regularizing parameters, a greater number producing more edge cuts	[0.1–40]
L2	20		[5–40]
DEC_R1	0.1	Customizable node weight, value is the inverse of K1	
DEC_R2	0.1		
MAX_GAP	0.5	Maximally allowed threshold distance to consider an edge	[0.5–5]
VER_O_W	0.3	Ratio of elevation difference from neighbors to segment length	[0–1.1]
RHO	0.5	Importance of the horizontal overlapping ratio over the vertical	[0–2]

municipal forest (both situated in the federal state of Baden-Württemberg, Germany), and the Nagyerdő forest in Debrecen (located in the federal state of Hajdú-Bihar, Hungary) (Fig. 4). Terrain characteristics varied across the study sites. The Karlsruhe site (KA09) and Nagyerdő plots (A and B) were situated on flat terrain. In contrast, the two Bretten sites (BR01 and BR05) were characterized by a hilly landscape.

The German forests encompass a variety of main tree species, including Norway spruce (*Picea abies*), Scots pine (*Pinus sylvestris*), Douglas-fir (*Pseudotsuga menziesii*), common oak (*Quercus robur*), red oak (*Quercus rubra*), sessile oak (*Quercus petraea*), European beech (*Fagus sylvatica*), and European hornbeam (*Carpinus betulus*).

In KA09, Scots pine, red oak, and European beech were dominant. BR01 and BR05 exhibited a more diverse tree species composition, featuring spruce, Douglas-fir, European beech, various oaks, and European hornbeam. These forests are characterized by a mixed-species composition, dense canopy cover, and multiple layers. Detailed information on plot characteristics can be found in a previous study (Weiser et al. 2022). In each German plot, 30 tree segments were manually delineated using the ALS-derived vertical profiles (Fig. 4).

Acquisition flight for the German forests was carried out using a RIEGL VQ-780i (RIEGL Laser Measurements Systems, 2019) sensor. The flight was performed on 5 July 2019. As for Nagyerdő, the ALS data was acquired on 3 March 2023 using the RIEGL VQ-780ii system. For a detailed description of the German ALS acquisition, see Weiser et al. (2022). ALS data characteristics of the five study areas are summarized in Table 3.

In the Nagyerdő forest plots, a mix of native and cultivated tree species thrived. These include common oak, red oak, silver poplar (*Populus alba*), Austrian pine (*Pinus nigra*), Scots pine, common hackberry (*Celtis occidentalis*), and eastern American black walnut (*Juglans nigra*). Both sites exhibit dense vegetation and are predominantly characterized by broadleaf-dominated mixed-composition. Site A exhibited a heterogeneous two-layered structure with a closed canopy layer. In contrast, forest B displayed an uneven structure consisting of multiple layers, with overgrowth of European ivy (*Hedera helix*). Forest A contained 56 tree segments, whereas site B contained 30 trees (Fig. 4). These segments were carefully created by visual inspection of the vertical profiles obtained from the high-resolution ALS data.

Four consecutive laser scans were conducted over the Nagyerdő forest to create a high-resolution dataset. The acquisition took place during the leaf development phase (with young leaves growing), which we considered the leaf-off period. The achieved mean point density within forest A was 162 pts/m^2, while in forest B it was 217 pts/m^2 (Table 3).

After manually segmenting point clouds of individual trees from the ALS data (Bretten, Karlsruhe, and Debrecen), the segmented tree point clouds were used as inputs for *treecbh*. We term these data as "input point cloud." The input point cloud characteristics are summarized in Table 4.

TLS forest

In addition to the ALS dataset, a single TLS dataset was used. Our TLS forest, Wytham Woods, is located in Oxfordshire County, UK (Fig. 5). It is managed by the Oxford University (https://www.forestgeo.si.edu/sites/europe/wytham-woods). The dominant tree species in Wytham Woods are European Ash (*Fraxinus excelsior*), sycamore (*Acer pseudoplatanus*), and common hazel (*Corylus avellana*).

TLS data were collected under leaf-off conditions from late November 2015 to January 2016. All scans were performed on windless days. A summary of the main TLS characteristics is provided in Table 5. For more information, please refer to the study by Calders et al. (2022).

Thirty trees were selected using a random sampling method from the available individual-tree-based TLS data of Wytham Woods (Calders et al. 2022). These data served as inputs for the point density sensitivity analysis (described in Section Point density senitivity).

Point density sensitivity

The point density of the 30 input point clouds (trees) selected from the Wytham Woods TLS data (Section TLS forest) was randomly decimated using the *decimate_points()* function from the lidR package (Roussel et al. 2020). This function employs a sampling algorithm that randomly samples or removes points from the area of a tree point cloud to obtain the desired point density (pts/m^2). The following six point density classes were defined: 20, 53, 111, 156, 205, and 253 pts/m^2.The analysis was conducted by executing *treecbh* in its CBH-detect-only mode (*cbh_ONLY* =3 in Table 1, as described in Section Parameters of *treecbh*). Accordingly, the selected input point clouds of the Wytham Woods TLS data were processed using the CBH detection tool, considering the six point density classes (Fig. 6).

Reference data

CBHs of the trees were measured in Bretten and Karlsruhe-Hardtwald forests using a Haglöf Vertex-IV hypsometer during exhaustive field measurements (Weiser et al. 2022). The CBH was defined as the height of the lowest branch, with a minimum length of 1 m.

Regarding the Nagyerdő, reference CBHs were collected through a field survey of 86 input point clouds (outlined in Section

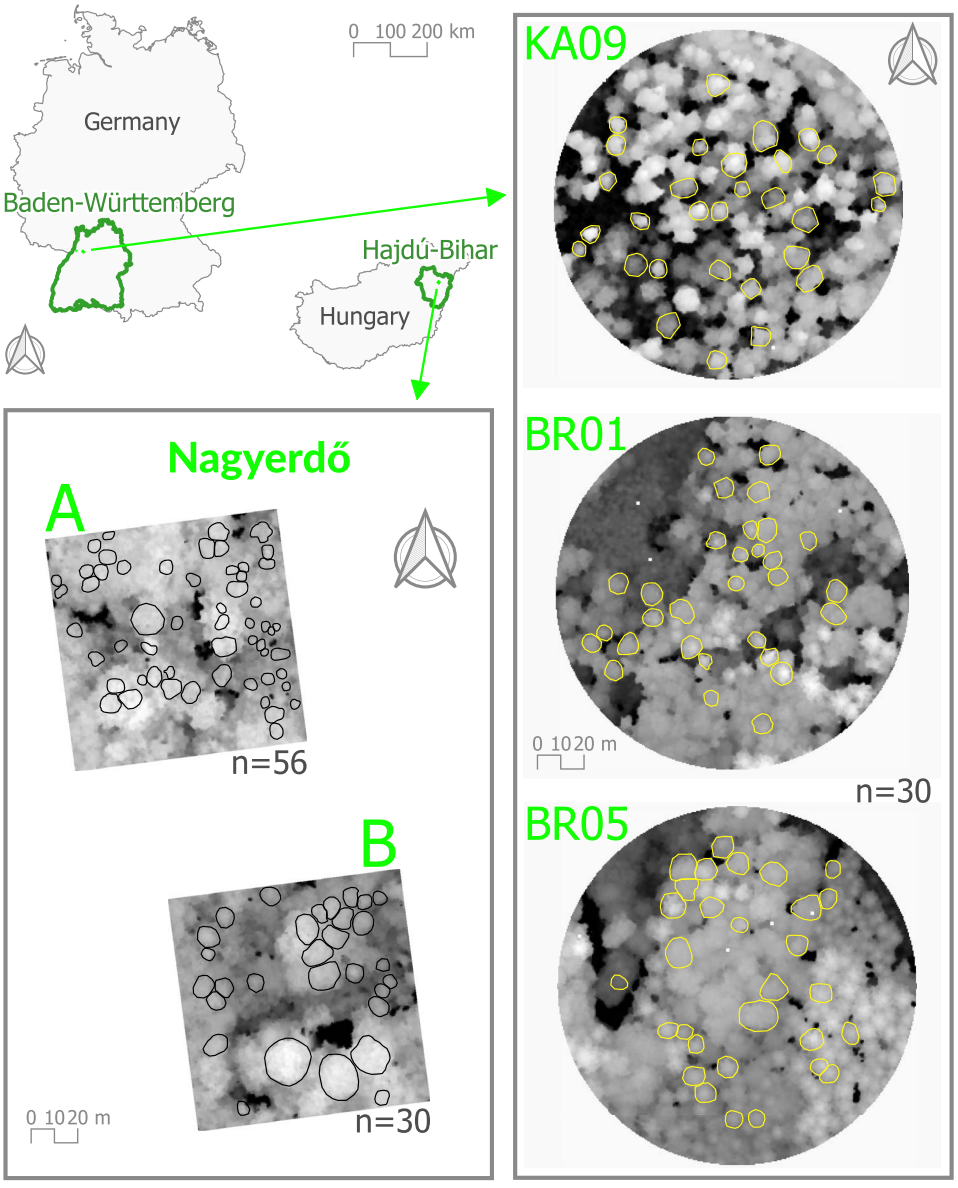

Figure 4. The three German plots (KA09, BR01, and BR05) are situated in the federal state of Baden-Württemberg, along with the two 1-ha sites (A and B) of the Nagyerdő mixed forest plantation in Hajdú-Bihar County. The number of tree segments utilized as input for *treecbh* is indicated by n=.

Table 3. Characteristics of the ALS data and their derived point clouds

Characteristics	BR01, BR05, and KA09			A and B	
ALS sensor	RIEGL VQ-780i			RIEGL VQ-780ii	
Flying altitude above ground level	650 m			664 m	
Off-nadir scan angle	±30°			±30°	
Laser beam divergence	0.25 mrad			0.18 mrad	
Pulse repetition frequency	1000 kHz			2000 kHz	
Flight line distance	175 m			210 m	
Flight line overlap	76%			60%	
Plot	BR01	BR05	KA09	A	B
Input point clouds' mean point density (pts/m^2)	204	172	186	162	217

ALS forests and tree segments) in forests A and B. The heights of the first leaved branches were measured in the field using the Haglöf EC II D-R electronic tool (https://haglofsweden.com/project/ec-ii-d-r). Trees were marked at 2 m with a measuring tape to verify the accuracy of the tool and ensure the precision and reliability of the measurements.

CBH was defined as the height of the first leaved branch protruding from the stem. The measured CBH value was

Table 4. Input point cloud characteristics

Plot	Number of input point clouds	Mean height (m)	Conifer–broadleaf ratio*
A	56	23	0.25:0.75
B	30	24	0.1:0.9
BR01	30	33	0.4:0.6
BR05	30	35	0.6:0.4
KA09	30	28	0.6:0.4

*Calculated from inventory data.

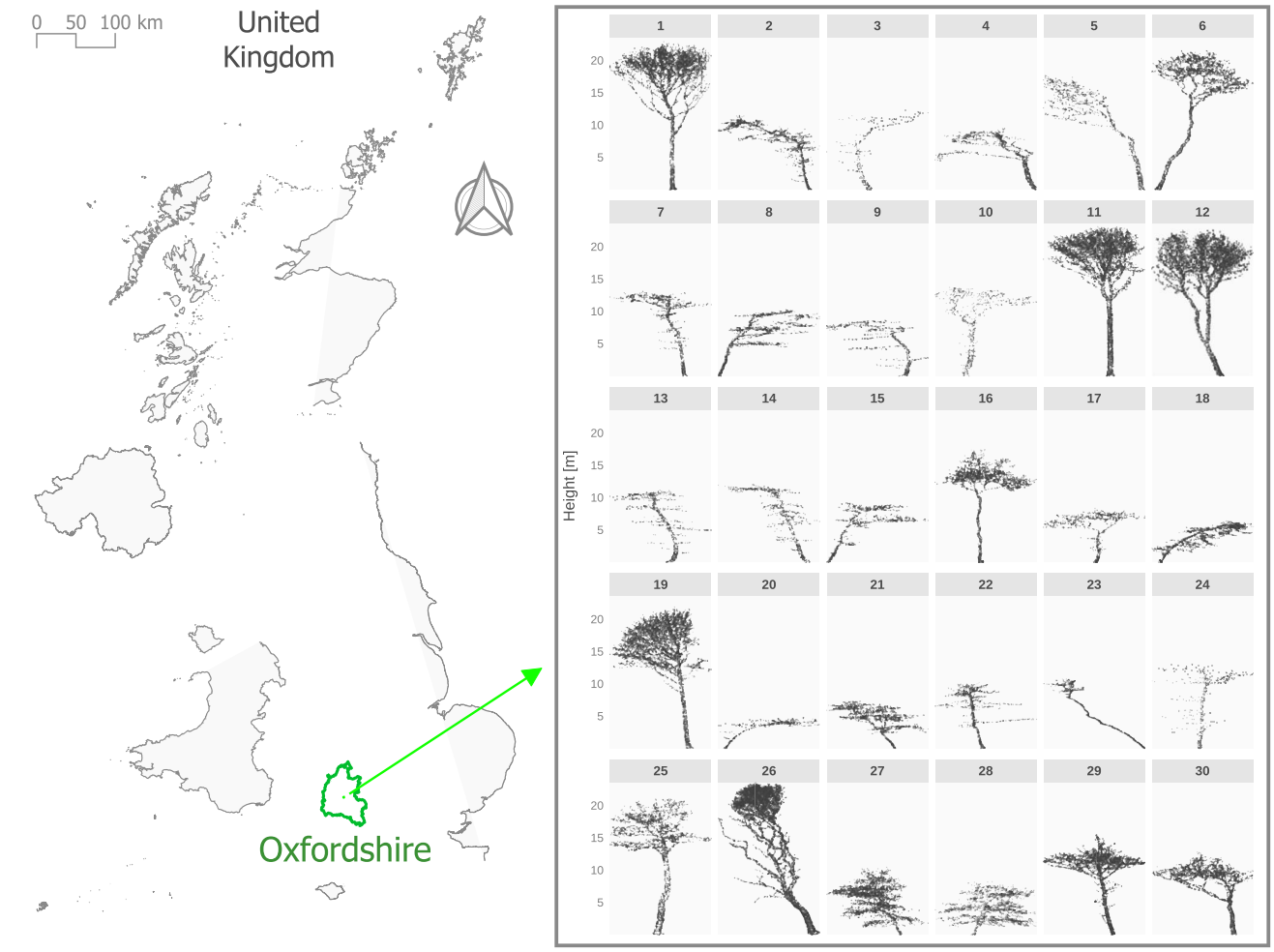

Figure 5. Location of the TLS plot in Wytham Woods shown on the left and the 30 randomly selected trees displayed in 2D on the right.

Table 5. TLS data characteristics

Characteristics	
TLS	**RIEGL VZ-400**
Minimum range	0.5 m
Maximum range	350 m
Laser beam divergence	0.35 mrad
Pulse repetition frequency	300 kHz
Angular sampling resolution	0.04°
Outgoing pulses per scan	22 500 000
Beam diameter	2.45 cm
Input point clouds' mean point density (Fig. 5)	4000 pts/m^2 → 111 pts/m^2*

*111 pts/m^2 was the mean point density of the finalized input point clouds.

determined based on the reading displayed by the EC II D-R tool when aimed at a precise spot on the tree stem where the first leaved branch initiated its protrusion. It is important to note that the measurement of the reference CBH did not account for the width or direction (downward and upward) of the respective tree branch.

Reference CBHs for the input point clouds of the TLS data were created through visual interpretation as the very high point density of the data (Table 5) enabled obtaining CBHs accurately and no field-measured data were available for this dataset.

Accuracy assessment

The detected CBHs were overlaid onto vertical 2D cross-sectional plots of the input point clouds (Section ALS forests and tree

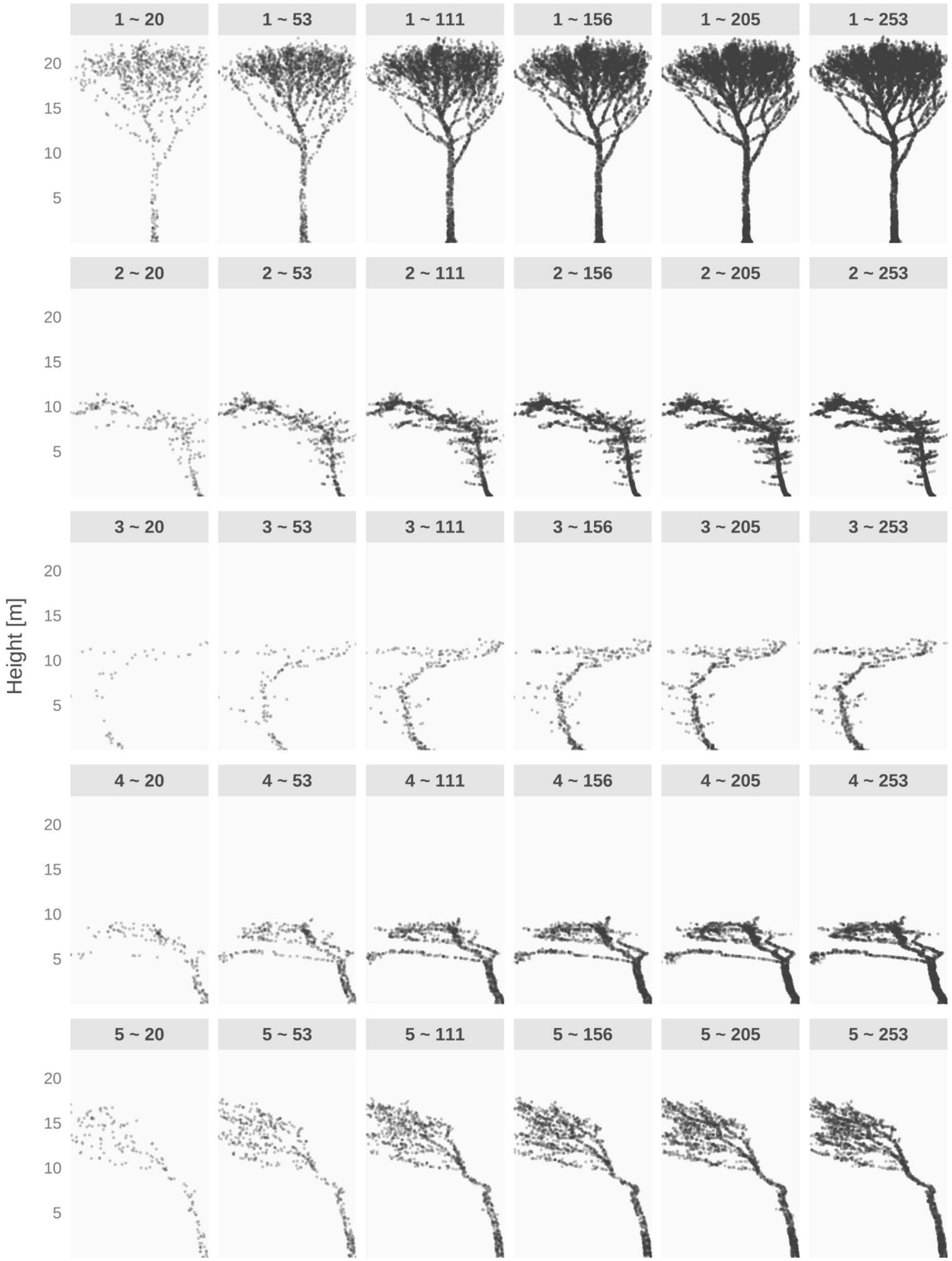

Figure 6. The first five tree point clouds (Fig. 5) from the selected Wytham Woods TLS data are displayed with respect to the six point density classes used for the point density sensitivity analysis: 20, 53, 111, 156, 205, and 253 pts/m^2.

segments). CBHs were buffered by 1 m in both the positive and negative directions along the *z*-axis (Fig. 1, Crown base height and Fig. 3). A matching criterion was applied to assess the match between the estimated and the buffered reference CBHs. A match (+) was recorded if the detected CBH fell within the buffered CBH range; otherwise, it was considered a non-match (–). The

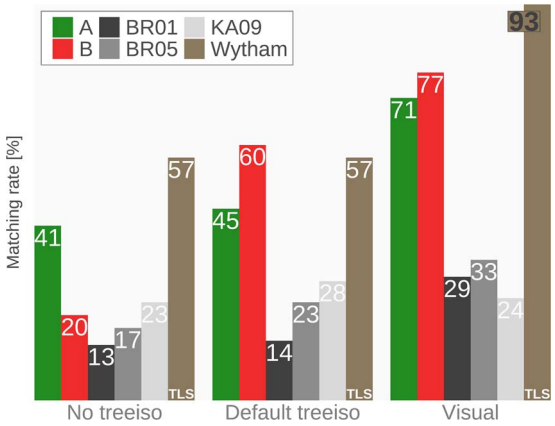

Figure 7. Comparison of crown base height detection results from ALS sites A, B, BR01, BR05, KA09, and TLS site Wytham Woods across three operating modes of the *treecbh* tool: deactivated *treeiso* (*No treeiso*), default parameter treeiso (*Default treeiso*), and interactive visual (*Visual*) CBH adjustment mode.

matching rate was calculated using Equation 2:

$$Matching\ rate_{[\%]} = (1 - (N_{(+)}/N_{(-)})) \cdot 100 \quad (2)$$

where the two *N*s represent the counts (i.e. sums) of matches (+) and non-matches (−), respectively. The mean absolute error (MAE) was applied to quantify the detected CBH deviation from the reference CBH and was computed using Equation 3:

$$MAE_{CBH\ [m]} = (1/N) \cdot \sum |Ref_{CBH_i} - Test_{CBH_i}| \quad (3)$$

where *i* indicates the *i*-th individual tree segment, and Ref_{CBH} and $Test_{CBH}$ are the reference and the detected CBHs, respectively. Parameter *N* represents the number of tree segments and the modulus is denoted by the | | symbol. The unit of MAE is given in meters, as indicated by the small *m* in square brackets.

The accuracy assessment consists of three sequential steps. First, the CBH detector was executed utilizing the input point clouds (Section ALS forests and tree segments) employing the deactivated *treeiso* mode (*No treeiso*). Next, *treecbh* estimated the CBHs for the input point clouds in conjunction with the activated *treeiso* mode. The parameters of the tree isolator were maintained at their default settings (Section Parameters of *treeiso*) and were denoted as *Default treeiso* throughout this study. Finally, the input point clouds were processed using *treecbh*'s interactive visual CBH adjustment mode, herein referred to as the *Visual* mode.

Results

Matching rate

At site per treecbh mode

At sites A and B (Hungary), we achieved matching rates of 41% and 20% in *No treeiso* mode, which were augmented to 45% and 60%, respectively, with the *Default treeiso*. Notably, the interactive visual CBH adjustment further increased these rates to 71% and 77% (Fig. 7).

For the three German forests, *treecbh* produced low matching rates of 13–23% in *No treeiso*, 14–28% in *Default treeiso*, and 24–33% in *Visual* mode (Fig. 7). However, what stands out is the consistent enhancement observed across *treecbh*'s operating modes in the BR05 forest. Notably, in the *Visual* mode, BR05 achieved a matching rate of 33%, which was the highest among German forests (Fig. 7).

As expected, the CBH detection of the TLS input point clouds was not influenced by activating *treeiso* since the input TLS trees were isolated in their default state. Both No treeiso and Default treeiso modes yielded a 57% matching rate. Both *No treeiso* and *Default treeiso* modes yielded a 57% matching rate (Fig. 7).

Comparison of leaf-off and leaf-on data

In the *Default treeiso* mode, the CBH detector attained a matching rate of 60% in forest B, identifying 18 matches and 12 non-matches (Fig. 8). The distinct panels in Fig. 8 illustrate that the structural information of the tree was effectively captured by leaf-off input point clouds.

In the BR05 German forest, the *Default treeiso* mode of *treecbh* achieved a matching rate of 23%, with 7 matches and 23 non-matches (Fig. 9). It is important to emphasize the insufficient representation of forest structure by input point clouds, as illustrated in Fig. 9. Laser scanning during the leaf-on season resulted in the majority of laser pulses being reflected by the canopy leaves, hindering effective penetration. As a result, tree structures under and within the canopy were not captured, contrary to what was clearly observed in forest B (Fig. 8).

MAE at site per treecbh mode

In Nagyerdő forests A and B, MAEs of 2.6 and 3.7 m were revealed for the *No treeiso* mode. Upon implementing tree isolation, these errors were reduced to 2.1 and 1.5 m (Fig. 10), respectively. Adjustment of CBH in the *Visual* mode resulted in a significant reduction in errors, with MAEs decreasing to 1.1 and 0.8 m (Fig. 10).

In comparison, the three German forests exhibited high MAEs across all three CBH detector operating modes (Fig. 10). It is evident that *treecbh* cannot handle the airborne laser scanning data scanned in the German forests.

Point density sensitivity

The performance evaluation of *treecbh*, focusing solely on CBH detection, revealed the influence of the input point cloud densities on the results. Contrary to expectations, increasing the point densities above 111 pts/m^2 did not enhance *treecbh*'s performance (Fig. 11). The corresponding performance remained at a 57% matching rate, as shown in Fig. 7. Indeed, the 111 pts/m^2 point density aligns with that employed for CBH detection on the TLS input point clouds (Table 5).

While *treecbh* demonstrated a 20% matching rate when processing input point clouds with a density of 20 pts/m^2, when presented with higher point densities of 156, 205, and 253 pts/m^2, matching rates ranging from 40% to 43% were achieved (Fig. 11).

Discussion

Performance of *treecbh*

Our findings highlight that the performance of our proposed CBH detector is primarily influenced by the quality of the input data, specifically the ability of the input point clouds to accurately describe the structure of trees under or within the canopy (Figs. 7 and 11). Before utilizing *treecbh*, we recommend that users assess the input point cloud and verify whether the input point cloud sufficiently represents the tree structure, particularly the tree trunk and lowest branches. This is a basic requirement for the algorithm to achieve meaningful results.

Given that leaf-on scanning tends to produce high point densities within the canopy (Fig. 9), but limited densities for the lower parts of the canopy, leaf-on data are less suitable as input to *treecbh*. The *treecbh*'s CBH detection technique relies on points

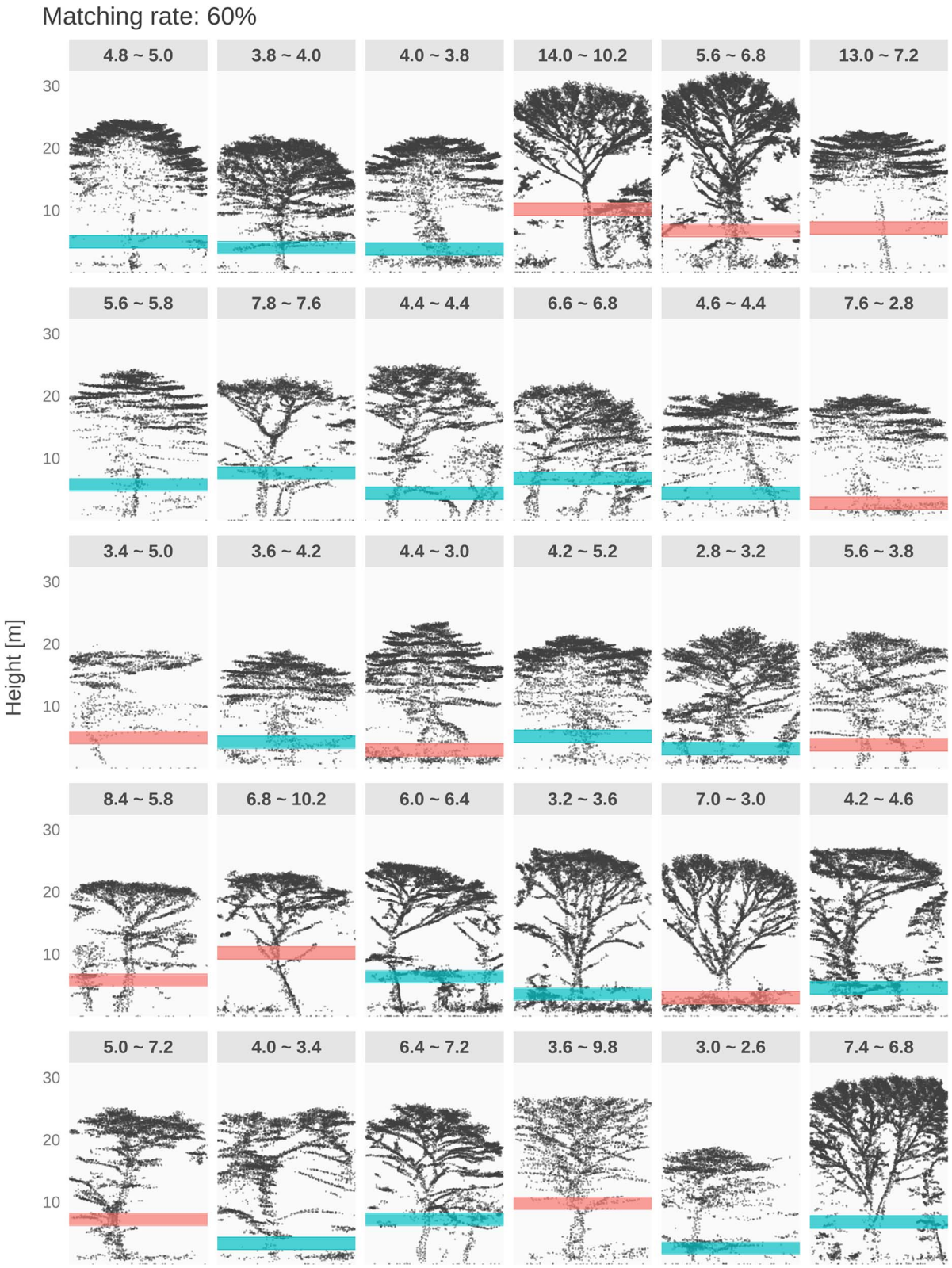

Figure 8. Crown base height detection results of *treecbh*: matching rate of the leaf-off ALS site B in *Default treeiso* mode. Horizontal transparent red bands (2 m wide) indicate non-matches, whereas transparent green bands (2 m wide) represent matches. Each panel is labeled with the reference CBH and its corresponding estimated CBH separated by the ~ symbol.

representing the tree trunk and lower branches (as detailed in Section The *treecbh* tool), and it is therefore imperative that the structure of trees beneath the canopy is adequately captured by the input point data. Therefore, the preferred input to *treecbh* is leaf-off ALS data (Fig. 8). Moreover, to effectively utilize *treecbh*'s *Visual* mode, ensuring the "readability" of tree structure (i.e. tree

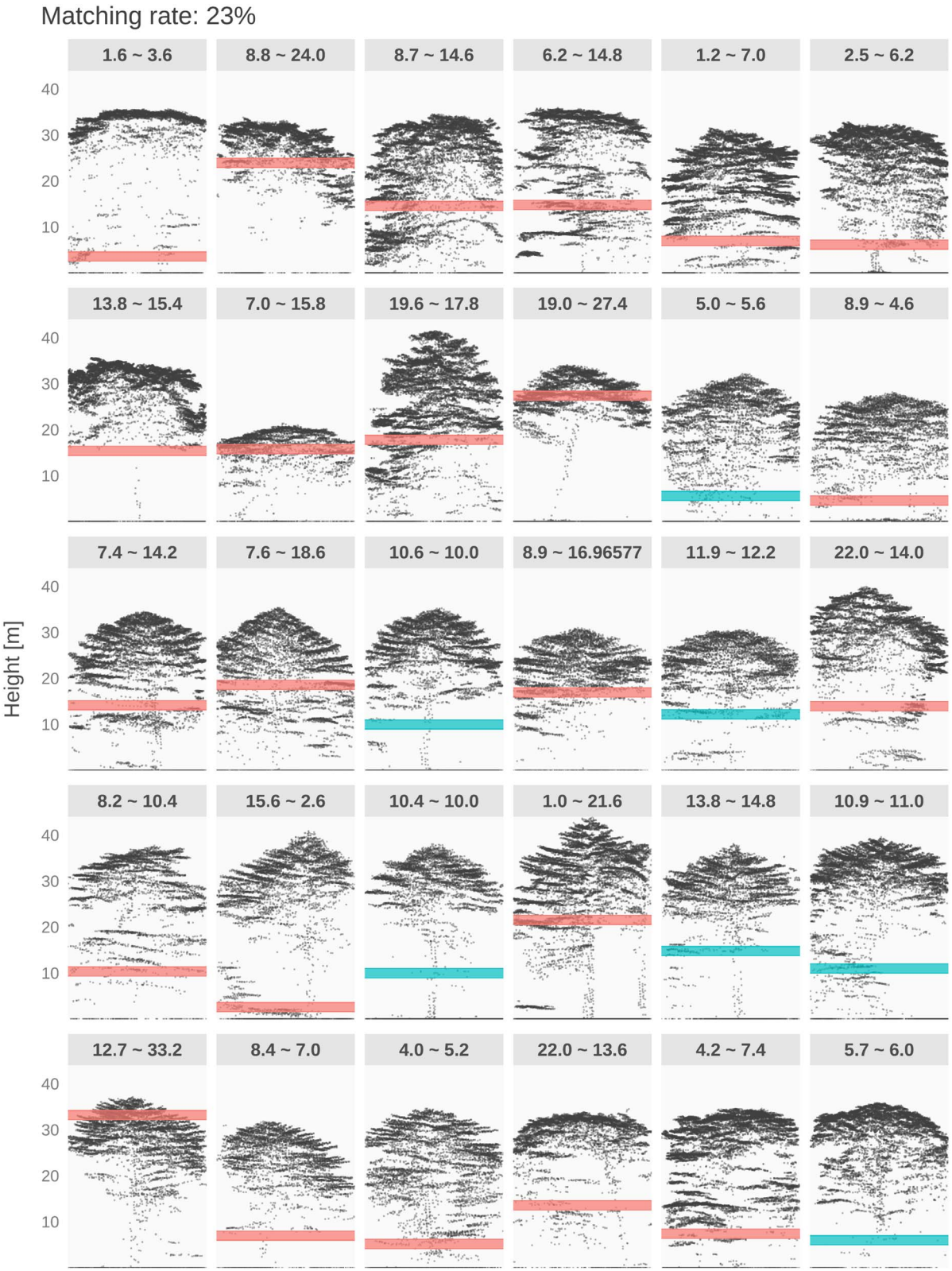

Figure 9. Crown base height detection results of *treecbh*: matching rate of the leaf-on ALS site BR05 in *Default treeiso* mode. Horizontal transparent red bands (2 m wide) indicate non-matches, whereas transparent green bands (2 m wide) represent matches. Each panel is labeled with the reference CBH and its corresponding estimated CBH, separated by the ~ symbol.

trunk and its lowest branches) is essential. Users must be able to reliably identify the CBH in order to provide the assumed CBH during the interactive visual CBH adjustment process (Fig. 3).

The *treecbh* tool demonstrated its optimal performance using the leaf-off ALS point cloud of forest B (Fig. 7), which had a conifer–broadleaf ratio of 0.1:0.9 (Table 4). In contrast, for the

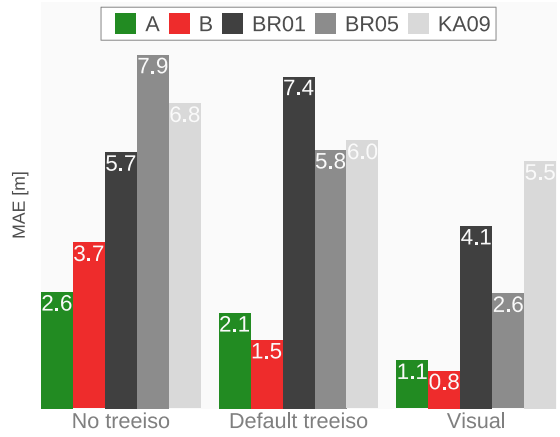

Figure 10. Crown base height detection results of *treecbh*: mean absolute errors of ALS sites A, B, BR01, BR05, and KA09 regarding *treecbh*'s three operating modes: deactivated *treeiso* (*No treeiso*), default parameter *treeiso* (*Default treeiso*), and interactive visual (*Visual*) CBH adjustment mode, respectively.

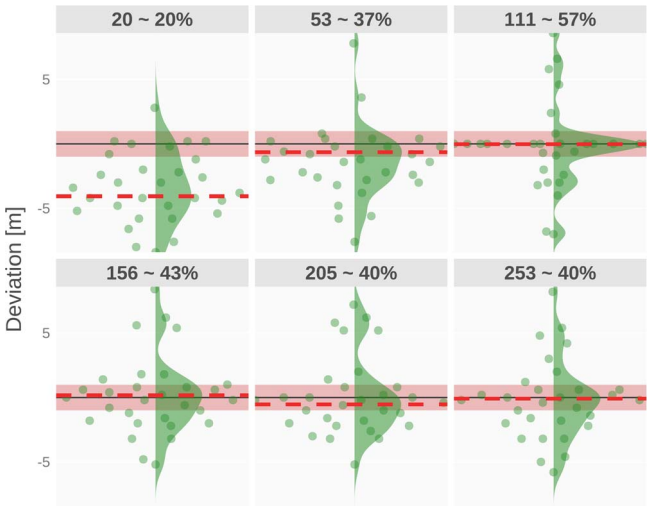

Figure 11. Crown base height detection results of point sensitivity analysis (*No treesio* mode) for point density classes. The horizontal dark line represents the reference CBHs, and the horizontal dashed lines mark the maxima of the deviation densities, which represent the distributions of the 30 input point clouds (dots). Horizontal transparent band (2 m wide) marks the ±1 m acceptance region. Each panel is labeled with its point density class along with its respective matching rate separated by the ~ symbol.

point cloud of forest A, which was characterized by a conifer–broadleaf ratio of 0.4:0.6, *treecbh* showed reduced robustness (Fig. 7). Furthermore, the lowest performance was observed using the point cloud from the leaf-off BR01 forest, which had a conifer–broadleaf ratio of 0.6:0.4 (Table 4). These findings underscore the robustness of our CBH detector in broadleaf forests compared with mixed conifer forests. Thus, our CBH detector is limited to performing well in broadleaf forests, and we recommend using the current version of the detector in such a setting.

Interestingly, the performance of *treecbh* in the *No treeiso* mode improved only when the point density increased from 20 to 111 pts/m^2 (Fig. 11). This suggests that a minimum point density is required for the sound operation of *treecbh*. A point density of ~100 pts/m^2 appears to be adequate for effectively representing tree structure. To evaluate *treecbh*'s CBH detection, we used a mean point density of 111 pts/m^2 for the TLS point cloud (Table 5 and Fig. 7). This density was chosen because it proofed to visually well represent the tree structure. However, it is important to note that the TLS data collected during leaf-off conditions may be perceived as less realistic when assuming an operational scenario as compared to ALS datasets from Nagyerdő, Bretten, and Karlsruhe forests, as the TLS data provided perfectly delineated individual tree point clouds.

Future development of *treecbh*

Addressing the inherent structural intricacies within the point clouds is crucial. The vertical arrangement of a forest (single, double, or multi-layered) significantly influences the outcomes of treetop detection and crown segmentation (Eysn et al. 2015). Consequently, *treecbh* faces specific challenges in under-canopy environments, particularly in isolating tree stems in the lower canopy and removing understory vegetation.

To isolate tree stems effectively in the lower canopy, an additional algorithm is necessary, e.g. the two-stage cut-pursuit clustering algorithm (Xi and Hopkinson 2022), which was implemented in this study. Enhancing the performance of our CBH detector should start with the implementation of a more robust tree isolator, such as 3DFin, which is available as a CloudCompare plugin as well as a Python module (Laino et al. 2024).

Another important aspect for the future development of *treecbh* involves the identification and effective removal of understory vegetation (Luo et al. 2018). This task needs to be carried out at the individual tree level, especially in multispecies heterogeneous forest stands.

In its current implementation, *treecbh* utilizes the *minH* formula (Eq. 1) to remove understory vegetation. Its scaling parameter, *min_H_scale*, was consistently set at 0.13 throughout this study (Section Parameters of *treecbh*). Although adjusting *min_H_scale* has demonstrated potential for enhancing CBH detection accuracy, determining its optimal value necessitates iterative testing. We refrained from doing so in this study, but future work could focus on this, as well as refining the *minH* formula to enhance the accuracy of CBH detection.

Operational use of *treecbh*

When compared to previous studies that focused on CBH estimation at the individual tree level, our CBH detector is competitive only when its *Visual* mode is utilized on leaf-off data, achieving MAE values of 0.8 and 1.1 m. Vauhkonen (2010) reported a root mean square error (RMSE) between 1.54 and 3.56 m for Scots pine (Vauhkonen 2010). In a similar study that estimated CBH for both broadleaf and conifer species, an average RMSE of 2 m was obtained (Popescu and Zhao 2008). Similarly, an RMSE of ~1.6 m has been achieved in mixed-species forests (Luo et al. 2018). These studies used ALS data of 4 to 10 pts/m^2 point densities, scanned in structurally simpler forests compared to our study areas. Tree isolation and understory removal have not been reported to be particularly challenging for these forests (Luo et al. 2018; Popescu and Zhao 2008). Addressing these challenges (Section Future development of *treecbh*) was a key driver in the development of *treecbh*.

In our study, we used manually extracted tree segments to generate individual tree input point clouds (Section ALS forests and tree segments). For the operational implementation of our CBH detector, this segmentation step must precede the use of *treecbh*. The second crucial step involves employing a tree isolator capable of effectively identifying the tree trunk and lower branches within the input point cloud. To address potential inaccuracies in CBH detection owing to imperfect tree isolation, *treecbh* offers a visual

CBH adjustment mode. However, relying solely on the workflow as outlined in this study is suboptimal for operational viability, partly because of the manual segmentation process and partly because of reliance on visual adjustment.

Alternatively, achieving high CBH detection accuracy in the *Default treeiso* mode is feasible if the tree isolator used is sufficiently robust (as described in Section Future development of *treecbh*) to handle inaccuracies such as the presence of other trees or parts of neighboring trees in the input point cloud. Implementing a more robust tree isolator would facilitate automation of the initial segmentation process. Automating this segmentation process using any suitable segmentation algorithm would significantly enhance the operational viability of *treecbh*.

Conclusions

We developed a comprehensive framework for tree isolation, understory removal, and CBH detection from leaf-off ALS data. Our tool, *treecbh*, is implemented as an R package and seamlessly integrates with the *lidR* framework. The tool is equipped with fixed default parameters, enabling straightforward execution while providing users with an option for visual CBH detection adjustment. An innovative aspect of this adjustment process is its interactive nature, where *treecbh* engages with the user. Our investigations revealed that when evaluated against field-measured reference data, the default parameter set yielded CBH matching rates of 45%–60% for leaf-off ALS data. However, these rates were outperformed by visual CBH adjustment, which achieved matching rates of 71%–77%.

Users are advised to use leaf-off ALS data and become familiar with the quality of the point clouds and their information content before utilizing *treecbh*. This familiarity helps decide whether to use the interactive visual CBH adjustment feature. Additionally, because the tree isolator, *treeiso*, is integrated within *treecbh*, users can perform only tree isolation by deactivating CBH detection. Notably, users can also expedite the visual CBH adjustment process by deactivating the tree isolator.

As we look ahead, our future endeavors involve enhancing *treecbh*'s performance by replacing *treesio* with a more robust tree isolator.

Acknowledgements

We are grateful to Envirosense Hungary Ltd. for providing the ALS data used in this study.

Author contributions

Gergő Diószegi: conceptualization, methodology, software, visualization, validation, writing—original draft preparation. Vanda Éva Molnár: data collection and editing. Lóránd Attila Nagy: data collection. Péter Enyedi: providing data, data curation. Péter Török: reviewing and editing. Szilárd Szabó: writing—reviewing and editing, supervision.

Conflict of interest: None declared.

Funding

The authors were supported by the NKFI K138079 and the KKP 144068 projects during manuscript preparation.

Data availability

The package *treecbh* is available at https://github.com/DijoG/treecbh and can be installed by executing the following line:

devtools::install_github('DijoG/treecbh').

The scanned ALS data from the three German plots used in this study can be downloaded from https://doi.pangaea.de/10.1594/PANGAEA.942856.

The TLS point clouds of the individual trees from Wytham Woods are available at https://zenodo.org/records/7307956.

References

Andersen H-E, McGaughey RJ, Reutebuch SE. Estimating forest canopy fuel parameters using LIDAR data *Remote Sens Environ*. 2005;**94**:441–9. https://doi.org/10.1016/j.rse.2004.10.013.

Bianchi S, Siipilehto J, Hynynen J. How structural diversity affects Norway spruce crown characteristics *For Ecol Manage*. 2020;**461**:117932. https://doi.org/10.1016/j.foreco.2020.117932.

Botequim B, Fernandes P, Borges J. *et al.* Improving silvicultural practices for Mediterranean forests through fire behaviour modelling using LiDAR-derived canopy fuel characteristics *Int J Wildland Fire*. 2019;**28**:823. https://doi.org/10.1071/WF19001.

Calders K, Verbeeck H, Burt A. *et al.* Laser scanning reveals potential underestimation of biomass carbon in temperate forest *Ecol Solut Evid*. 2022;**3**:e12197. https://doi.org/10.1002/2688-8319.12197.

Dean TJ, Cao QV, Roberts SD. *et al.* Measuring heights to crown base and crown median with LiDAR in a mature, even-aged loblolly pine stand *For Ecol Manage*. 2009;**257**:126–33. https://doi.org/10.1016/j.foreco.2008.08.024.

Engelstad PS, Falkowski M, Wolter P. *et al.* Estimating canopy fuel attributes from low-density LiDAR *Fire*. 2019;**2**:38. https://doi.org/10.3390/fire2030038.

Erdody TL, Moskal LM. Fusion of LiDAR and imagery for estimating forest canopy fuels *Remote Sens Environ*. 2010;**114**:725–37. https://doi.org/10.1016/j.rse.2009.11.002.

Eysn L, Hollaus M, Lindberg E. *et al.* A benchmark of lidar-based single tree detection methods using heterogeneous forest data from the alpine space *Forests*. 2015;**6**:1721–47. https://doi.org/10.3390/f6051721.

Finney MA. An overview of FlamMap fire modeling capabilities. Andrews Patricia Butl. Bret W comps 2006 fuels Manag.- Meas. Success conf. Proc. 28-30 march 2006 Portland proc. RMRS-P-41 Fort Collins CO US dep *Agric For Serv Rocky Mt Res Stn*. 2006;213–220 041.

Finney MA. FARSITE: Fire area simulator-model development and evaluation. Res pap RMRS-RP-4 Revis. 2004 Ogden UT US dep. Agric. For. Serv. Rocky Mt. res. Stn. 47 P 4 1998. https://doi.org/10.2737/RMRS-RP-4.

Girardeu-Monteau D. CloudCompare, v2.13 alpha 2023.

González-Ferreiro E, Diéguez-Aranda U, Crecente-Campo F. *et al.* Modelling canopy fuel variables for Pinus radiata D. Don in NW Spain with low density LiDAR data *Int J Wildland Fire*. 2013;**23**:350. https://doi.org/10.1071/WF13054.

Hermosilla T, Ruiz LA, Kazakova AN. *et al.* Estimation of forest structure and canopy fuel parameters from small-footprint full-waveform LiDAR data *Int J Wildland Fire*. 2014;**23**:224–33. https://doi.org/10.1071/WF13086.

Hsu W-C, Shih PT-Y, Chang H-C. *et al.* A study on factors affecting airborne LiDAR penetration *Terr Atmospheric Ocean Sci*. 2015;**26**:241. https://doi.org/10.3319/TAO.2014.12.02.08(EOSI).

Jakubowksi MK, Guo Q, Collins B. *et al.* Predicting surface fuel models and fuel metrics using lidar and CIR imagery in a dense, mountainous forest *Photogramm Eng Remote Sens*. 2013;**79**:37–49. https://doi.org/10.14358/PERS.79.1.37.

Kelly M, Su Y, Di Tommaso S. *et al.* Impact of error in lidar-derived canopy height and canopy base height on modeled wildfire behavior in the sierra Nevada, California, USA *Remote Sens (Basel)*. 2018;**10**:10. https://doi.org/10.3390/rs10010010.

Korhonen L, Vauhkonen J, Virolainen A. *et al.* Estimation of tree crown volume from airborne lidar data using computational geometry *Int J Remote Sens*. 2013;**34**:7236–48. https://doi.org/10.1080/01431161.2013.817715.

Laino D, Cabo C, Prendes C. *et al.* 3DFin: A software for automated 3D forest inventories from terrestrial point clouds *For Int J For Res*. 2024;**97**:479–96. https://doi.org/10.1093/forestry/cpae020.

Landrieu, Obozinski. Cut pursuit: Fast algorithms to learn piecewise constant functions. In: *Proceedings of the 19th International Conference on Artificial Intelligence and Statistics*. Presented at the Artificial Intelligence and Statistics, PMLR, 2016, 1384–93.

Luo L, Zhai Q, Su Y. *et al.* Simple method for direct crown base height estimation of individual conifer trees using airborne LiDAR data *Opt Express*. 2018;**26**:A562–78. https://doi.org/10.1364/OE.26.00A562.

Maguya AS, Tegel K, Junttila V. *et al.* Moving voxel method for estimating canopy base height from airborne laser scanner data *Remote Sens (Basel)*. 2015;**7**:8950–72. https://doi.org/10.3390/rs70708950.

Næsset E, Økland T. Estimating tree height and tree crown properties using airborne scanning laser in a boreal nature reserve *Remote Sens Environ*. 2002;**79**:105–15. https://doi.org/10.1016/S0034-4257(01)00243-7.

Popescu SC, Zhao K. A voxel-based lidar method for estimating crown base height for deciduous and pine trees *Remote Sens Environ*. 2008;**112**:767–81. https://doi.org/10.1016/j.rse.2007.06.011.

Riaño D, Meier E, Allgöwer B. *et al.* Modeling airborne laser scanning data for the spatial generation of critical forest parameters in fire behavior modeling *Remote Sens Environ*. 2003;**86**:177–86. https://doi.org/10.1016/S0034-4257(03)00098-1.

Roussel J-R, Auty D, Coops NC. *et al.* lidR: An R package for analysis of airborne laser scanning (ALS) data *Remote Sens Environ*. 2020;**251**:112061. https://doi.org/10.1016/j.rse.2020.112061.

Stefanidou A, Gitas IZ, Korhonen L. *et al.* LiDAR-based estimates of canopy base height for a dense uneven-aged structured Forest *Remote Sens (Basel)*. 2020;**12**:1565. https://doi.org/10.3390/rs12101565.

Sumnall M, Fox T, Wynne R. *et al.* Mapping the height and spatial cover of features beneath the forest canopy at small-scales using airborne scanning discrete return lidar *ISPRS J Photogramm Remote Sens*. 2017;**133**:186–200. https://doi.org/10.1016/j.isprsjprs.2017.10.002.

Sumnall M, Peduzzi A, Fox TR. *et al.* Analysis of a lidar voxel-derived vertical profile at the plot and individual tree scales for the estimation of forest canopy layer characteristics 2016.

Terryn L, Calders K, Åkerblom M. *et al.* Analysing individual 3D tree structure using the R package ITSMe *Methods Ecol Evol*. 2023;**14**:231–41. https://doi.org/10.1111/2041-210X.14026.

Tibshirani R, Walther G. Cluster validation by prediction strength *J Comput Graph Stat*. 2005;**14**:511–28. https://doi.org/10.1198/106186005X59243.

Vauhkonen J. Estimating crown base height for scots pine by means of the 3D geometry of airborne laser scanning data *Int J Remote Sens*. 2010;**31**:1213–26. https://doi.org/10.1080/01431160903380615.

Vauhkonen J, Ene L, Gupta S. *et al.* Comparative testing of single-tree detection algorithms under different types of forest *For Int J For Res*. 2012;**85**:27–40. https://doi.org/10.1093/forestry/cpr051.

Weiser H, Schäfer J, Winiwarter L. *et al.* Individual tree point clouds and tree measurements from multi-platform laser scanning in German forests *Earth Syst Sci Data*. 2022;**14**:2989–3012. https://doi.org/10.5194/essd-14-2989-2022.

Xi Z, Hopkinson C. 3D graph-based individual-tree isolation (Treeiso) from terrestrial laser scanning point clouds *Remote Sens (Basel)*. 2022;**14**:6116. https://doi.org/10.3390/rs14236116.

Xu W, Su Z, Feng Z. *et al.* Comparison of conventional measurement and LiDAR-based measurement for crown structures *Comput Electron Agric*. 2013;**98**:242–51. https://doi.org/10.1016/j.compag.2013.08.015.

Zarnoch S, Bechtold W, Stolte K. Using crown condition variables as indicators of forest health *Can J For Res*. 2004;**34**:1057–70. https://doi.org/10.1139/x03-277.