

RESEARCH

Open Access



# Individual tree extraction from terrestrial laser scanning data via graph pathing

Di Wang<sup>1</sup>, Xinlian Liang<sup>2,3\*</sup>, Gislain Il Mofack<sup>4</sup> and Olivier Martin-Ducup<sup>5</sup>

## Abstract

**Background:** Individual tree extraction from terrestrial laser scanning (TLS) data is a prerequisite for tree-scale estimations of forest biophysical properties. This task currently is undertaken through laborious and time-consuming manual assistance and quality control. This study presents a new fully automatic approach to extract single trees from large-area TLS data. This data-driven method operates exclusively on a point cloud graph by path finding, which makes our method computationally efficient and universally applicable to data from various forest types.

**Results:** We demonstrated the proposed method on two openly available datasets. First, we achieved state-of-the-art performance on locating single trees on a benchmark dataset by significantly improving the mean accuracy by over 10% especially for difficult forest plots. Second, we successfully extracted 270 trees from one hectare temperate forest. Quantitative validation resulted in a mean Intersection over Union (mIoU) of 0.82 for single crown segmentation, which further led to a relative root mean square error (RMSE%) of 21.2% and 23.5% for crown area and tree volume estimations, respectively.

**Conclusions:** Our method allows automated access to individual tree level information from TLS point clouds. The proposed method is free from restricted assumptions of forest types. It is also computationally efficient with an average processing time of several seconds for one million points. It is expected and hoped that our method would contribute to TLS-enabled wide-area forest qualifications, ranging from stand volume and carbon stocks modelling to derivation of tree functional traits as part of the global ecosystem understanding.

**Keywords:** Point cloud, Segmentation, Tree extraction, Graph pathing

## Background

Past two decades have witnessed the great advancements of terrestrial laser scanning (TLS) technique in delineating the unprecedented three-dimensional (3D) structures of trees and forests (Calders et al. 2020). By far, TLS provides the most accurate measures at the scale of individual trees through its product named point cloud (Liang et al. 2016). A point cloud contains highly precised 3D coordinates in combination with radiometric information of objects (Pfeifer et al. 2008). A range of applications including

retrieval of conventional forest inventories parameters such as stem location (Liang et al. 2014), diameter at the breast height (DBH) (Wang et al. 2017), stem curve (Wang et al. 2019), and estimate of tree physiological properties such as leaf angle distribution (Liu et al. 2019), branching architecture (Lau et al. 2018), and standing volume (Calders et al. 2015) have shown the promising capability of TLS in quantifying ecosystem structure and function.

A key prerequisite to these applications is the proper interpretation of the information content of point clouds (Burt et al. 2019). In particular, individual trees have to be extracted first from plot level or wider area data to enable the retrieval of interested parameters at tree-scale. For example, newly developed shape-fitting approaches such as quantitative structure model (QSM) allow the 3D

\*Correspondence: [xinlian\\_liang@hotmail.com](mailto:xinlian_liang@hotmail.com)

<sup>2</sup>State Key Laboratory of Information Engineering in Surveying Mapping and Remote Sensing, Wuhan University, 430079 Wuhan, China

<sup>3</sup>Department of Remote Sensing and Photogrammetry, Finnish Geospatial Research Institute, 02430 Masala, Finland

Full list of author information is available at the end of the article

reconstruction of tree woody structures (Raumonen et al. 2013; Hackenberg et al. 2015; Du et al. 2019). Based on QSMs, tree above-ground volume, biomass, carbon stocks, branching structures (Lau et al. 2018), and also species (Åkerblom et al. 2017) can be accurately estimated and recognized. Such applications all require the point cloud of individual trees to be identified first. Currently, this task is often carried out by laborious and time-consuming manual manipulations, which is impractical for large-area data delineation. An automated solution with high effectiveness and efficiency is greatly desirable.

Over the past years, a number of approaches were developed to tackle the task of individual tree extraction from TLS data. Most methods followed a bottom-up mentality by first identifying tree bases or stems, then gradually allocating crown points to respective bases. In earlier developments, only tree stems were extracted using methods such as feature-based filtering (Liang et al. 2014), Euclidean clustering (Hackenberg et al. 2015), circle fitting (Pueschel et al. 2013), and point cloud projection (Wang et al. 2016). Consequently, less attentions were paid on detailed crown segmentation. Nevertheless, several methods were proposed, especially in past five years, to focus on crown segmentation. Raumonen et al. (2015) proposed a morphological approach that locates tree stem first. Then detected stems were expanded to crowns using fixed distance connectivity assumptions. The same idea was later implemented by Trochta et al. (2017). Limitation of this method is its restricted applicability on open forests with sparsely distributed trees only (Burt et al. 2019). A popular group of methods used graph based theory and technique. Tao et al. (2015) was among the firsts to apply graph shortest path algorithm to resemble the nature of trees for transporting water and nutrients in a most efficient way. Wang (2020) also used the same principle by constructing a superpoint graph instead, and achieved single tree extraction and leaf-wood separation simultaneously. Zhong et al. (2016) utilized normalized graph cut as an energy minimization technique to group crown points to candidate stems. Similarly, Heinzl and Huber (2018) developed a graph-based constrained spectral clustering approach which minimized the energy of a Markov Random Field. In a recent development, deep learning based approach has been explored for tree extraction as well (Wang et al. 2019). In general, all these methods demand the tree bases or stems to be identified first prior to graph-based manipulations. Moreover, some automatic methods required manual assistance either in field (Barbeito et al. 2017) or in post-processing (Calders et al. 2015; Burt et al. 2019). In addition, quantitative assessment of point-wise crown segmentation was only carried out for synthetic data before (Wang 2020). The available datasets with ground-truth crown boundaries are still very rare nowadays.

The objective of this study is to present a fully automatic approach for individual tree extraction from TLS data. Specifically, we propose a graph pathing algorithm that operates on the point cloud graph only. We jointly search for tree bases or roots and allocate other points to respective roots within the graph by using graph path finding techniques. In this way, we also ensure a fast computation even for wide-area data. Moreover, this study aims to provide the first quantitative assessments of point-wise crown segmentation in real forests.

## Materials

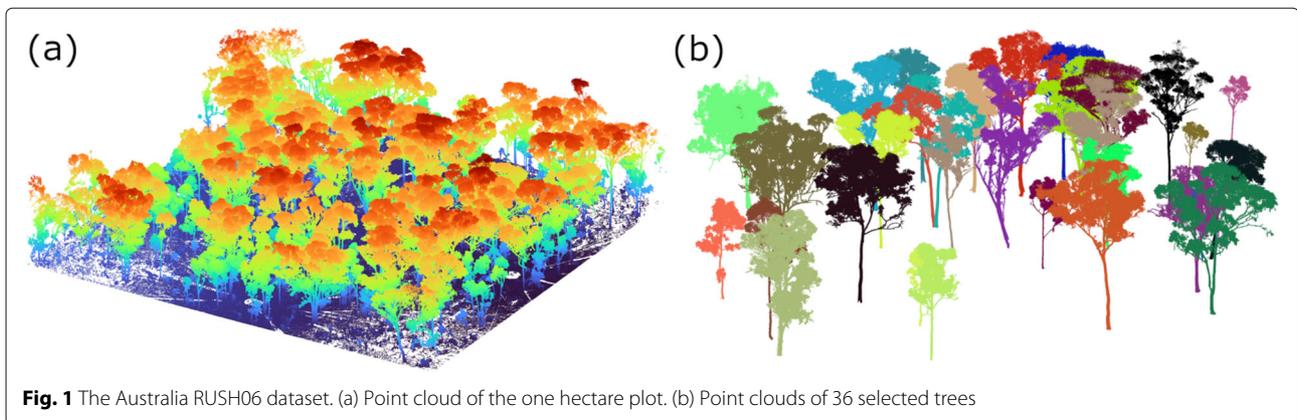
The assessments of the effectiveness of single tree extraction involve two metrics, tree location and detailed crown segmentation. Among them, tree location can be manually measured in the field or identified in the point clouds, whereas single tree crowns are very challenging to delineate even visually (Heinzl and Huber 2018). Therefore, most studies on single tree extraction evaluated their methods on tree locations (i.e., stem mapping) only. The assessment of point-wise crown segmentation has rarely been materialized. To the best of our knowledge, there exists no open datasets with both ground-truth tree locations and detailed crown boundaries, except synthetic ones (Wang 2020). On account of this, we used two separate datasets in this study to evaluate our method on locating single trees and isolating crowns, respectively. One dataset from Finland was included to benchmark our study with others on stem mapping. Additionally, we included a large-area dataset from Australia that provided selected manually extracted single trees to assess our method on crown segmentation.

### Finland data

We acquired openly available TLS datasets of six forest plots from the international TLS benchmarking project, which was organized by the Finish Geospatial Research Institute (FGI) (Liang et al. 2018). We provide here an overview of this dataset, and the readers are referred to Liang et al. (2018) for a full description of these benchmarking plots.

Overall, these datasets were acquired from southern boreal forests in Evo, Finland. The dominant tree species were Scots pine (*Pinus sylvestris* L.), Norway spruce (*Picea abies* L. Karst.), silver birch (*Betula pendula* Roth), and downy birch (*Betula pubescens* Ehrh.). The openly available six plots covered various tree species, stem densities, developing stages, and abundance of understory vegetation. These characteristics were further used to categorize the plots into three complexity types (i.e., easy, medium, and difficult) by the benchmarking organizer FGI.

The Leica HDS1600 scanner (Leica Geosystems AG, Heerbrugg, Switzerland) was used to obtain the point cloud data of each 32 × 32 m plot. Multi-scan mode was



deployed in the field, and both single- and multi-scan data were provided. Since we focus on full-tree extraction with a specific interest on detailed crown segmentation, only multi-scan data were used in this study. Moreover, we filtered out terrain points and normalized the height of each point to its above-ground value by using the open-source cloth simulation method (Zhang et al. 2016).

#### Australia data

The Australia data were gathered and shared by Calders et al. (2015). These data were collected from a native Eucalypt temperate Forest (dry sclerophyll Box-Ironbark forest) in Victoria, Australia. We acquired a subset dataset RUSH06 from the Rushworth forest, which had a stem density of 347 stems per hectare and a basal area of 13 m<sup>2</sup> per hectare, respectively.

TLS data were collected using the Riegl VZ-400 scanner (Riegl Laser Measurement Systems GmbH). The scan sampling setting deployed one center scan and four other scans at 40 m from the center. In this study, we cropped the plot to one hectare (100 × 100 m) (Fig. 1a), which contains approximately 37.5 million points. The study of (Calders et al. 2015) aimed for estimating tree above ground biomass from TLS data. A number of selected trees were thus destructively harvested and their

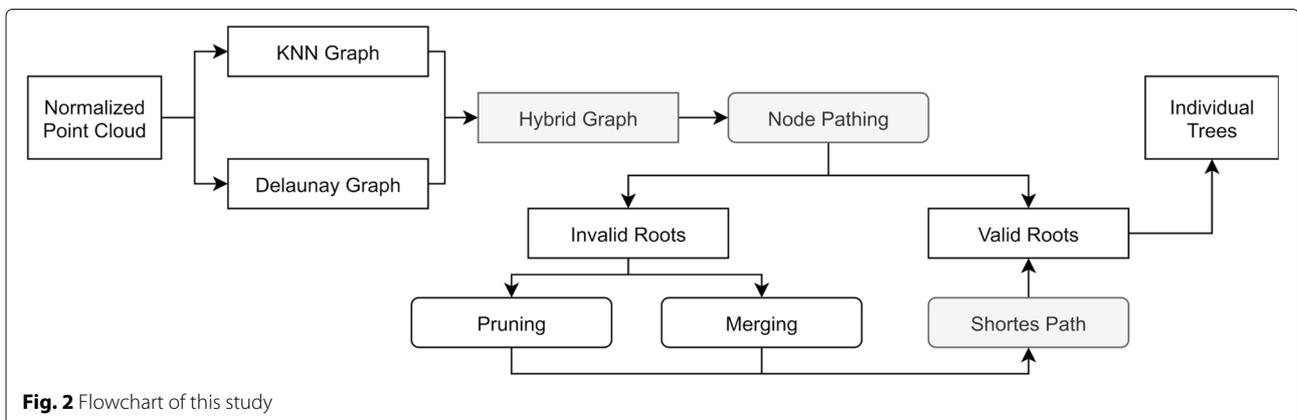
biomasses were precisely measured. The authors then semi-automatically extracted corresponding tree point clouds and reconstructed individual trees to compare the estimated biomass with destructive ones. In total, 36 trees were included from the RUSH06 forest (Fig. 1b). Point clouds of these trees were manually corrected and visually inspected. Consequently, they were used in this study to assess our results on detailed crown segmentation.

#### Methods

The proposed approach operates exclusively on the point cloud graph. The key idea is to walk each node through the graph edges and search for the correct root nodes. We use the height information that is inherently embedded in the node features to find valid root nodes. Ultimately, all nodes that arrive to the same root node belong to the same cluster (i.e., tree). The entire flowchart is illustrated in Fig. 2. We describe below in details the key steps in constructing the point cloud graph, pathing through the graph, and determining the node clusters.

#### Graph construction

A point cloud graph  $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{W})$  consists of a set of vertices  $\mathcal{V}$  (i.e., nodes), edges  $\mathcal{E}$  representing a set of



unordered pairs of vertices, and edge weights  $\mathcal{W} : \mathcal{E} \in \mathbb{R}$  mapping the Euclidean distances between graph nodes.

Conventionally, there exists several methods to convert a point cloud into a graph. The most widely used  $k$ -nearest neighbors (KNN) graph connects a point to its neighbors to generate a graph. Similarly, the neighboring connectivity can otherwise be defined by bounding points in a given radius  $R$ . On the other hand, such connectivity property can be directly modeled by the Delaunay triangulation. Each method has certain advantages and drawbacks. For example, KNN generates a balanced graph, but is vulnerable to distanced neighbors with dispersed edges. Moreover, KNN may be trapped to locally dense regions and results in a disconnected graph.  $R$  radius graph is robust to distanced neighbors but is sensitive to heterogeneous point densities. Lastly, Delaunay triangulation produces a fully connected graph but is vulnerable to noise and distanced neighbors as well. Ben-Shabat et al. (2018) compared all these methods for constructing a graph while aiming for point cloud segmentation, and recommended KNN based on their results.

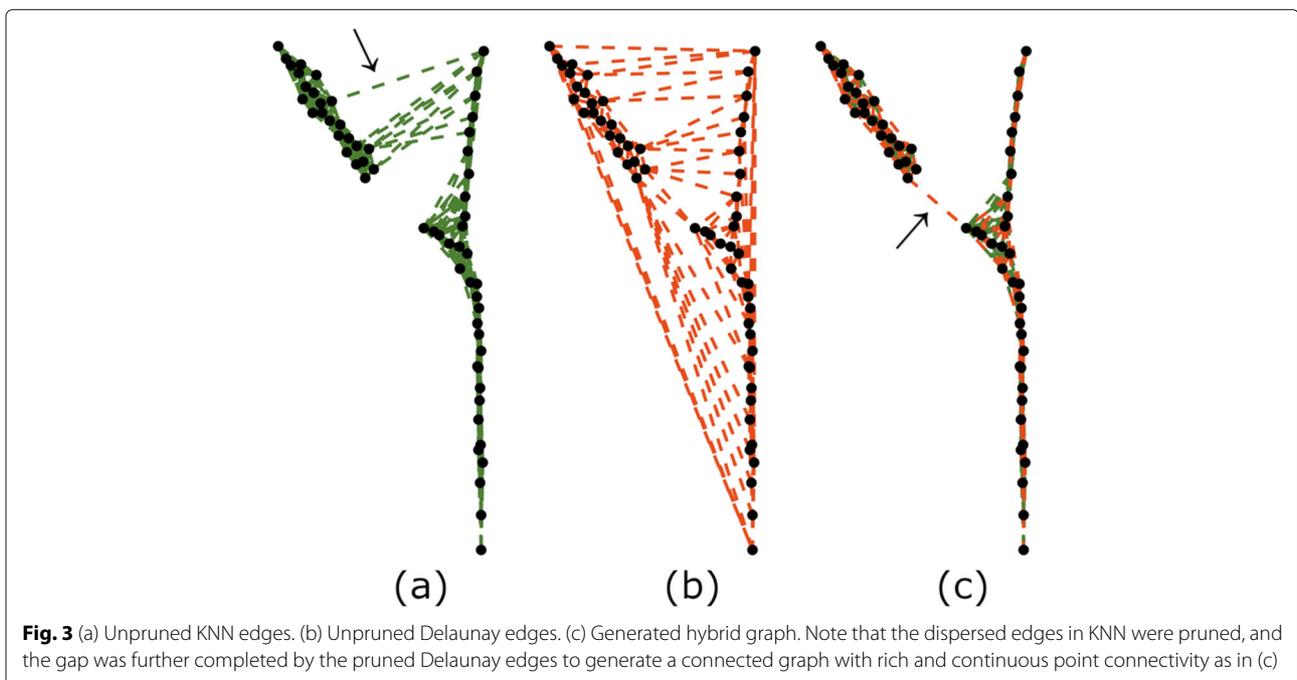
In our study, we leverage the advantages of KNN and Delaunay triangulation graph by merging them to generate a hybrid graph (Fig. 2). Specifically, we first construct a KNN (i.e.,  $k = 10$ ) graph based on the KD-Tree structure. We then prune the graph to remove dispersed edges. For each node, its edges that are one standard deviation anyway from the average length of edges connected to it are removed. In this way, the graph is locally pruned and optimized. Subsequently, the Delaunay triangulation is applied to generate a fully connected graph. Similarly, we

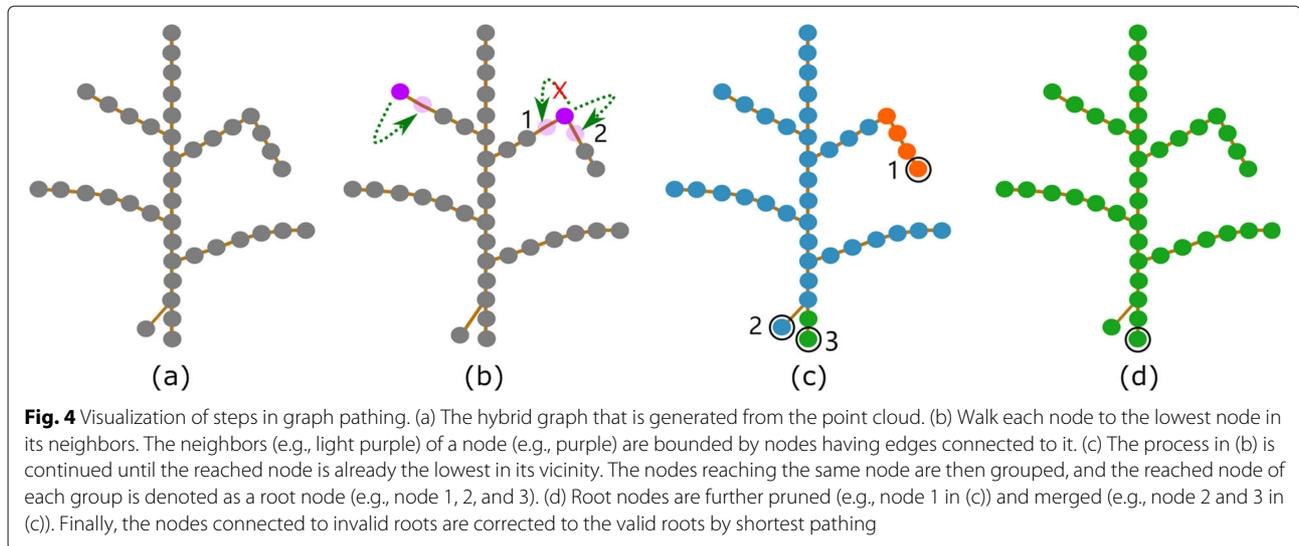
prune the Delaunay graph to remove long edges. However, the pruning is instead performed globally by excluding edges whose lengths are longer than the 80th percentile of all edge. A sensitivity analysis of these pruning criteria is given in the results section. Consequently, the pruned edges from both graphs are then merged. To do so, we achieve a hybrid graph with rich and continuous point connectivity by taking advantages of both techniques and avoided undesirable edges. Figure 3 illustrates the advantage of such as a hybrid graph.

### Node pathing

The key step of our method is to walk each node within the graph (i.e., pathing or path finding). Initially, we move each node to its lowest neighboring node (Fig. 4b) by assuming that the tree root node would have the lowest height. The neighboring relationship is bounded by graph edges. The reached lowest node in the neighbors then becomes the seed node, and this procedure is continued until the reached node cannot be moved anymore (i.e., the lowest in the vicinity). Accordingly, we achieve an initial clustering result, in which the nodes that reach the same lowest node are grouped (Fig. 4c). The reached lowest nodes are denoted as root nodes. Nevertheless, it is obvious that tree branching structures will not follow a rigid downward centripetal orientation. Some graph nodes will eventually land at their local lowest, rather than the global lowest which is expected to be the tree root node (Fig. 4c).

We then refine the detected root nodes to locate the desirable root nodes that represent tree roots. First, we prune root nodes that are higher than a threshold  $H$ . This





simple step already eliminates a majority of invalid root nodes. The determination of  $H$  depends on the quality of point clouds. For example, if a plot is well sampled by multi-scan TLS so that tree stems are clearly represented in the point cloud, the  $H$  value can be set to low such as 1 m. Otherwise,  $H$  can be increased to e.g., 3 m to mitigate the impacts from poorly sampled tree stems. Second, the remaining root nodes are further merged, as some nodes can be from the same tree stem. Specifically, we merge two root nodes if their Euclidean distance is shorter than a threshold  $Ed$ , and their graph distance is shorter than  $n$  (e.g., 3) times that of  $Ed$ . The graph distance is defined as the shortest path distance resolved by the Dijkstra algorithm (Dijkstra 1959). Indubitably,  $Ed$  is linked to the distance between two adjacent tree stems. Furthermore, the graph distance is a more robust measure than Euclidean distance in this circumstance. For example, two root nodes form two neighboring trees can be spatially near, but their graph distance is either infinite or very large. Therefore, by evaluating the graph distance, two spatially near root nodes can be further investigated if they are from the same tree or not.

Consequently, only valid root nodes are retained. All graph nodes that are initially landed at those invalid nodes are routed to a specific corresponding valid root node that has the shortest path to them (Fig. 4d).

### Node clustering

The above-mentioned pathing step allocates each graph node to a root, which is essentially a clustering procedure. Therefore, the outcome of this method is a number of point clusters that represent individual trees. It is noted that our method can be easily operated on a coarse point cloud first (e.g., superpoints), and the results can be mapped back to full resolution. In this study, we sampled

points from uniformly distributed 10 cm voxels to extract trees first, and the results are encoded with original resolution. This further accelerates the processing.

### Assessments

The assessments of our method is on two parts, tree locations and detailed crown segmentation.

Tree locations are assessed following the metrics used in the TLS benchmarking project (Liang et al. 2018). Three metrics including the completeness, the correctness, and the mean accuracy of detection are calculated. The completeness measures the percentage of detected reference trees. The correctness measures the percentage of detected trees against references. The mean accuracy is the joint metric based on the completeness and correctness, given by:

$$\text{Completeness} = \frac{n_{\text{match}}}{n_{\text{ref}}}, \quad (1)$$

$$\text{Correctness} = \frac{n_{\text{match}}}{n_{\text{extr}}}, \quad (2)$$

$$\text{Mean accuracy} = \frac{2n_{\text{match}}}{(n_{\text{ref}} + n_{\text{extr}})}, \quad (3)$$

where  $n_{\text{match}}$  is the number of detected reference trees,  $n_{\text{ref}}$  is the number of reference trees, and  $n_{\text{extr}}$  is the number of detected trees.

On the contrary, the Intersection over Union (IoU), a standard metric for segmentation evaluation measure, is used to assess the detailed crown segmentation. For an  $N \times N$  confusion matrix ( $N = 36$  in this study), each entry  $c_{ij}$  refers to the number of points from reference tree  $i$  predicted as tree  $j$ . Then the IoU of tree  $i$  is calculated as:

$$\text{IoU}_i = \frac{c_{ii}}{c_{ii} + \sum_{j \neq i} c_{ij} + \sum_{k \neq i} c_{ki}}. \quad (4)$$

The mean IoU (mIoU) of all trees is then estimated by:

$$mIoU = \frac{\sum_{i=1}^N IoU_i}{N}. \quad (5)$$

We additionally assessed our results for estimating crown area and tree volume. Crown area was calculated as the vertically projected bounding area of a tree, and tree volume was the volume of convex hull of the entire tree. The root mean square error (RMSE) and its relative value were reported as:

$$RMSE = \sqrt{\frac{1}{k} \sum_{i=1}^k (y_i - \hat{y}_i)^2}, \quad (6)$$

$$RMSE(\%) = 100\% \times \frac{RMSE}{\bar{y}}, \quad (7)$$

where  $k$  is the number of observation data,  $\hat{y}$  denotes the reference value and  $\bar{y}$  is the mean value of the variable.

### Method comparisons

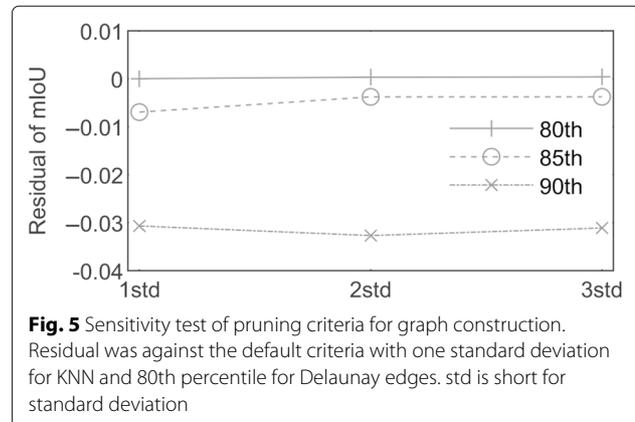
Several studies that also used the benchmark dataset from Finland reported their accuracies of locating single trees. We hereby performed quantitative comparisons of our results with two state-of-the-art approaches in Zhang et al. (2019) and Wang (2020). However, unfortunately, it is unfeasible to compare our results on detailed crown segmentation with both approaches, as only Wang (2020) was able to segment individual crowns, while Zhang et al. (2019) only detected tree locations.

## Results

### Sensitivity of pruning criteria for graph construction

In this study, we generated a hybrid graph by merging pruned KNN and Delaunay edges. The pruning operation involved two criteria, one standard deviation for KNN edges and 80th percentile for Delaunay edges, respectively. Theoretically, these criteria were only set to ensure a graph with rich and continuous point connectivity, and were not directly influential to the downstream analyses. Nonetheless, we provide here a sensitive analysis of these two pruning criteria for crown segmentation from the Australia dataset.

We tested the multiplication factor of standard deviation from one to three for KNN edges, and from 80th and 90th percentile with an increment of 5 percentile for Delaunay edges. In total, nine different combinations were tested. The results of mIoU for crown segmentation of the Australian dataset were shown in Fig. 5. Overall, the manipulations of pruning criteria did not impact the results significantly. The standard deviation of mIoUs from nine criteria combinations were only 1.5%. Specifically, the multiplication factor of standard deviation had negligible impacts. While by increasing the pruning threshold for Delaunay edges from 80th to 90th,



**Fig. 5** Sensitivity test of pruning criteria for graph construction. Residual was against the default criteria with one standard deviation for KNN and 80th percentile for Delaunay edges. std is short for standard deviation

the results worsened by about 0.03 in mIoU. This, however, was well expected as increasing this value meant that fewer long edges were removed. So the final graph was more vulnerable to dispersed edges from Delaunay triangulation. Overall, we showed that the pruning criteria for graph construction were insensitive.

### Finland data

The results of our method on locating single trees, and the comparisons with other two state-of-the-art approaches proposed in Zhang et al. (2019) and Wang (2020) were summarized in Table 1. Overall, our method outperformed others by a wide margin in terms of mean accuracy. The improvement was more significant for difficult plots with over 10% increase. Our method achieved 76.9% completeness for difficult plots, whereas Zhang et al. (2019) and Wang (2020) reported 36.1% and 58.3%, respectively. However, we were not able to compete with Zhang et al. (2019) on correctness. In general, our method resulted in a more balanced performance with the highest overall accuracy.

### Australia data

In total, we extracted 270 trees from the one hectare RUSH06 Australia data (Fig. 6). The mIoU was 0.82 for detailed crown segmentation, with a standard deviation of 0.14 and individual IoUs ranging from 0.51 to 1.00. As a comparison, Wang (2020) also achieved a mIoU of 0.82, but with a higher standard deviation of 0.19. As for the estimations of crown area and tree volume, our results showed an RMSE of 11.5 m<sup>2</sup> (21.2%) and 139.5 m<sup>3</sup> (23.5%), respectively (Fig. 7). The  $R^2$  of the linear regression line (Fig. 7a black line) for crown area estimation was 0.91, with a slope of 0.88. Similarly, the  $R^2$  of the linear regression line (Fig. 7b black line) for tree volume estimation was 0.91 as well. The slope was 0.95. These results indicate that our method is able to precisely segment individual crowns and is effective in estimating tree attributes

**Table 1** Tree extraction performance on Finland dataset

Plot Complexity	Completeness(%)			Correctness(%)			Mean accuracy(%)		
	Z	W	Ours	Z	W	Ours	Z	W	Ours
Easy	86.3	90.2	<b>96.1</b>	<b>97.8</b>	95.8	96.1	91.7	92.9	<b>96.1</b>
Easy	82.1	<b>92.9</b>	<b>92.9</b>	<b>95.8</b>	81.3	89.7	88.5	86.7	<b>91.2</b>
Medium	61.5	79.1	<b>84.5</b>	<b>100.0</b>	72.2	90.6	76.2	75.5	<b>87.4</b>
Medium	57.5	78.2	<b>89.7</b>	<b>97.8</b>	70.1	74.5	72.6	74.0	<b>81.4</b>
Difficult	45.8	64.1	<b>77.1</b>	<b>93.8</b>	67.2	68.7	61.5	65.6	<b>72.7</b>
Difficult	26.3	52.5	<b>76.7</b>	<b>98.4</b>	70.9	80.8	41.5	60.3	<b>78.7</b>
	59.9	76.2	<b>82.3</b>	<b>97.3</b>	76.3	83.4	72.0	75.8	<b>84.6</b>

Last row corresponds to mean values. Z: results from Zhang et al. (2019). W: results from Wang (2020)

associated to crowns. However, as shown in Fig. 8, accurate point-wise segmentation was always very challenging to achieve especially for heavily intersected crowns, for which the shortest path analysis generally failed (Wang 2020). Nonetheless, our results indicated that despite the challenge in point-wise segmentation, automatic crown-level metrics retrieval was instead reliable.

#### Method efficiency

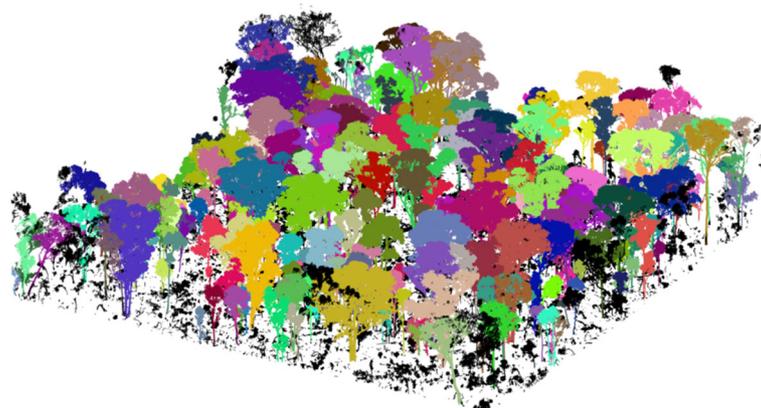
One of the advantages of our method is its computational efficiency. The runtime of our method depends on both total number of points and number of trees to be extracted (Fig. 9). In average, the processing time was 1.6 min for one plot from the FGI benchmark dataset (i.e., averaging 12.5 million points and 119 trees). For the one hectare Australia dataset with 37.5 million points, the runtime was 8.5 min and we managed to extract 270 trees. The machine we used to run the algorithm had the following specifications: Windows 10, Intel® Core™ i9-10900X and 32 GB RAM. Although different machines were used, our runtime is generally much lower than previous studies for full tree extraction (Burt et al. 2019; Wang 2020).

#### Discussion

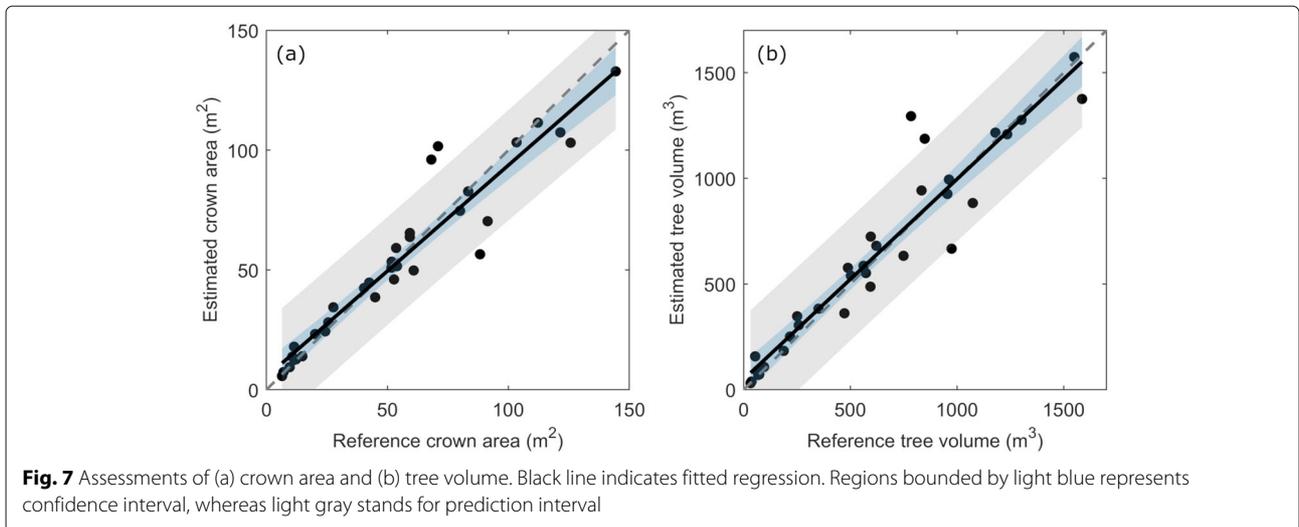
Although TLS had made promising progress in forest applications in past two decades, the automation of data processing remains a crucial challenge and very often time-demanding manual works are still needed (Calders et al. 2020). Meanwhile, a key prerequisite is to properly understand the information contents embedded in the forest plots so that individual tree level attributes can be retrieved (Burt et al. 2019). This requirement generally involves two fundamental tasks, single tree extraction and leaf-wood separation (Wang 2020). Our study focuses on automated individual tree extraction from large area data.

#### Challenges of tree extraction in complex forests

With regard to the practical application of a certain tree extraction method, forest types and data quality are two primary factors that defy its effectiveness, transferability, and universality. Most tree extraction methods developed for TLS data follow a bottom-up approach by first recognizing tree bases or stems, and then expanding the bases to branches to constitute the entire crowns. Obviously, this mentality coincides with the mechanism of



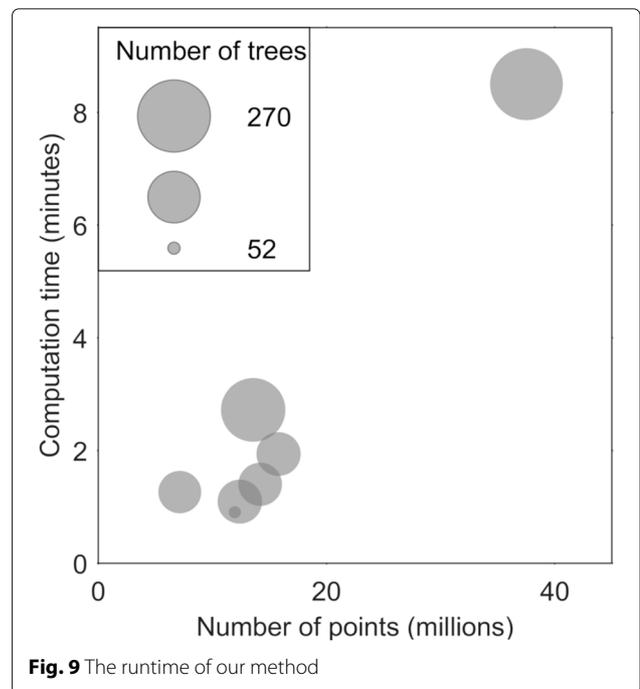
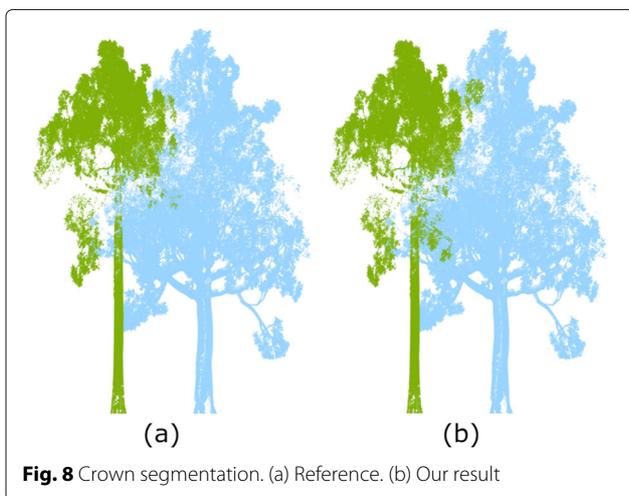
**Fig. 6** Extracted individual trees. Each tree is colored randomly, and black color indicates non-tree objects

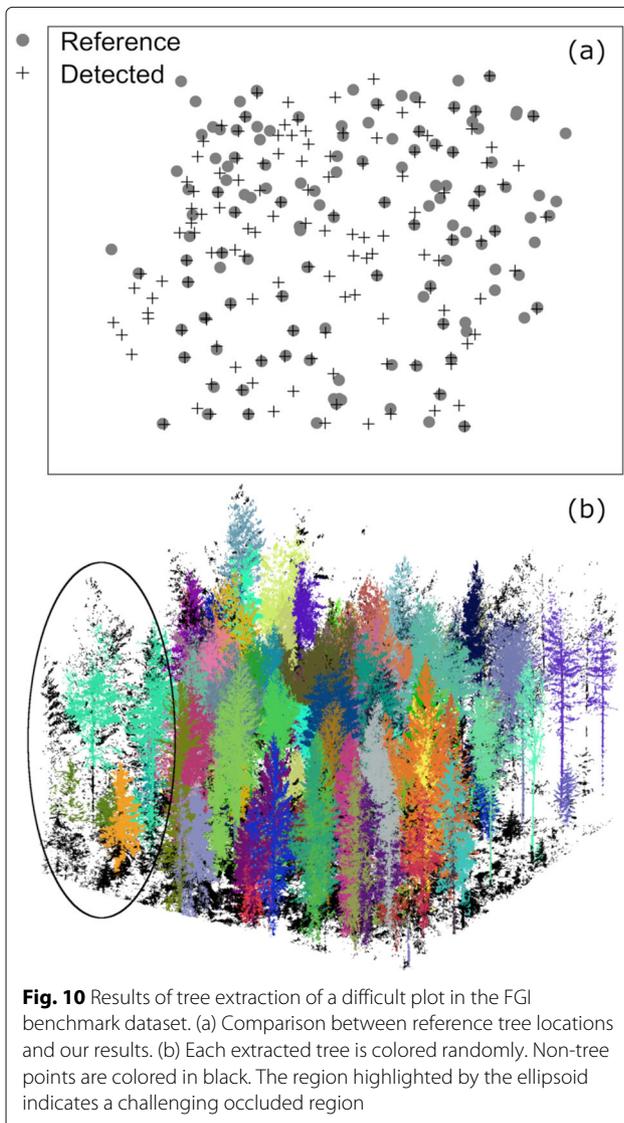


TLS measurements for acquiring data from near ground with an upward field of view. For structurally simple and sparse forest types with distanced trees and minimal interaction between crowns, it is plausible that many tree extraction methods are equally effective. However, more complex forests such as tropical rainforests and other natural ecosystems would impose significant challenges. For example, dense low vegetation and complex terrain conditions would greatly defect the data quality, as occlusions and limitations in setting up multi-scan TLS are often unavoidable. Consequently, the conventionally used clustering and primitive (e.g., circle or cylinder) fitting methods may fail in this circumstance when tree stems are not well resolved in the point clouds (Trochta et al. 2017; Tao et al. 2015; Burt et al. 2019). Thereby, many extraction methods are essentially semi-automatic and require manual assistance and quality control to correct omission or commission errors (Calders et al. 2020).

**Advantages of graph pathing**

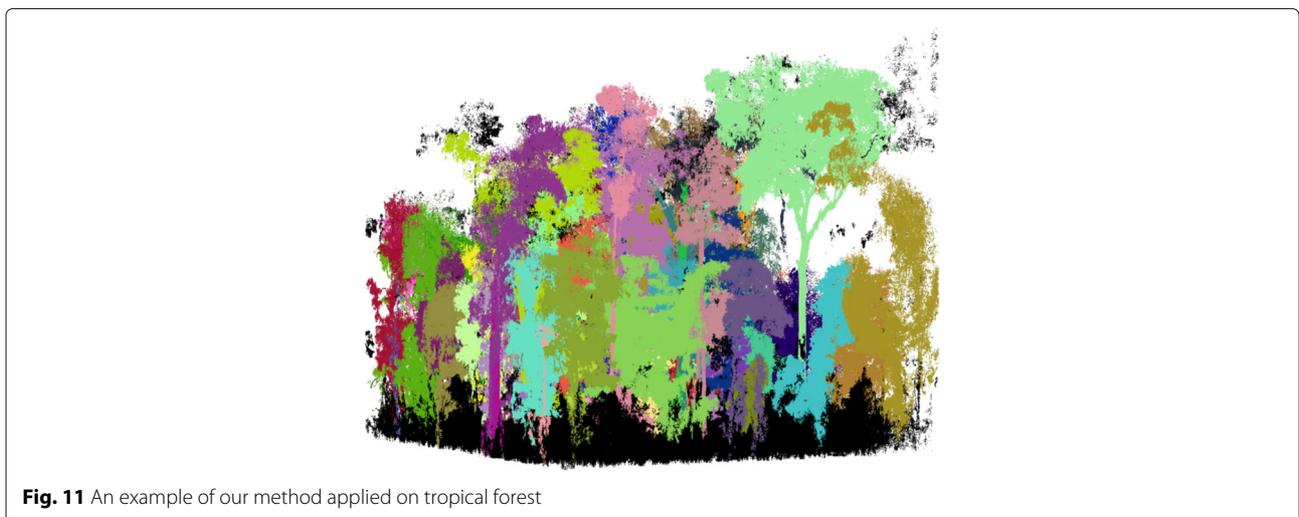
Our method operates exclusively on the point cloud graph and seeks for tree bases directly by node pathing, thus avoids point clustering or shape fitting that are more vulnerable to data quality. This advantage is demonstrated in our results for locating trees from the FGI benchmark dataset. We achieved much higher completeness (i.e., 76.9%) especially for difficult plots compared to other methods (Table 1), indicating that our method is more robust to occlusions and low points densities. However, we still noticed the impacts of occlusions in our results. In Fig. 10a, it was discovered that our method was able to





detect most trees in the central part of the plot, whereas missed a relatively large portion of trees on the plot edge. A visual inspection in Fig. 10b revealed that plot edges had severe occlusions with very sparse points. Several trees were mistakenly merged into one tree due to low confidences in locating tree bases. This impact may potentially be mitigated by leveraging the information from vertically stacked points (Wang et al. 2016), but would only be feasible for single layered forests. Overall, our method seems fall into the “Aggressive” algorithm principle according to the benchmarking summary, which strives to achieve the highest stem detection rate by sacrificing the correctness of stem detection (Liang et al. 2018). This was indeed revealed in our results for having much higher completeness compared to previous studies. However, the achieved state-of-the-art mean accuracy also indicated that our method was robust at the same time. It is noted that by tweaking the parameters such  $H$  and  $Ed$ , our method may turn into a more “Conservative” principle. As the benchmarking project concluded, the preference and selection of a specific algorithm all depend on the final objective of the applications (Liang et al. 2018). The flexibility of our method has the potential to ease such algorithm selection or design, thus facilitates its applicability in a wide variety of forest applications.

An additional advantage of our method is its capability to identify non-tree points that are linked to invalid roots (Figs. 2 and 6), which adds a particular value to tropical forests that have excessive undergrowth. A quick experiment on a  $40 \times 40$  m plot data from a tropical forest in Eastern Cameroon (Martin-Ducup et al. 2021) showed that around 30% points were attributed to non-tree objects such as shrubs (Fig. 11). Our method does not rely on point clustering or shape fitting, and it is free from assumptions such as tabular shapes when detecting stems,



which makes our method uniquely capable of segmenting and identifying undergrowth as well. Intrinsically, this approach is able to discover all possible roots including trees, shrubs, and other low vegetation. A large portion of undergrowth then can be easily identified by limiting their heights at the object level. This advantage makes our method particularly valuable in complex forests with dense undergrowth.

We have also shown that our method is able to precisely segment tree crowns. The yielded mIoU of 0.82 was comparable to a recent study from Wang (2020) on synthetic forests. Although it is expected that the precise isolation of tree crowns is extremely challenging especially in complex forests where trees have their crowns interact with each other, we showed that a fully automatic method can produce a reasonable segmentation that leads to a relative RMSE at the magnitude of 20% for crown area estimates. It is, however, remains unknown how such a fully automatic tree extraction would impact the estimation of more advanced tree physiological attributes such as tree leaf angle distribution (Liu et al. 2019) and above-ground biomass (AGB) (Calders et al. 2020).

Lastly, by restricting the computations on the graph only, we significantly reduced the computation overhead and runtime (Fig. 9). Currently, the efforts in collecting more and more TLS data from different geographical locations and ecosystems across the globe are rapidly accelerating, and a method that is computationally efficient and robust to different data characteristics is particularly wanted.

## Conclusions

Individual tree extraction from TLS data is a crucial and fundamental step in tree-scale analysis. In this study, we presented a simple and effective automatic approach for this task. The proposed method was exclusively based on a point cloud graph using path finding algorithms and avoided using conventional processing techniques such as point cloud feature calculation, clustering, and geometric primitive fitting that are more sensitive to data quality.

Experiment results showed that the proposed method achieved state-of-the-art performance on locating single trees when compared with previous studies. In particular, our method outperformed others by a wide margin in structurally complex forests. We also demonstrated that the proposed method was able to precisely segment tree crowns in large-area forests.

This study facilitates the application of TLS in forest research for which automatic and effective tree extraction in complex and large-area forests remains a crucial challenge. The proposed method is computationally efficient and can process one hectare forest with nearly 300 trees in 10 min with a standard machine. We argue that our method provides a significant advance in confronting the

new realm of untapped research questions and applications of TLS in forest ecosystems.

## Abbreviations

3D: Three dimensional; AGB: above-ground biomass; DBH: diameter at the breast height; FGI: Finish geospatial research institute; IoU: Intersection over Union; KNN: k-nearest neighbors; mIoU: mean Intersection over Union; QSM: quantitative structure model; RAM: Random access memory; RMSE: Root mean square error; TLS: Terrestrial laser scanning

## Acknowledgements

We are grateful to the FGI and Wageningen University (Dr. Kim Calders, now at Ghent University) for sharing TLS data.

## Authors' contributions

D. W, X.L. L, and O. MD together conceived the research idea and designed the experiments. D. W and G. M processed the data and analyzed the results. All authors contributed to the manuscript writing and editing. All authors read and approved the final manuscript.

## Funding

This work was partially funded by the Scientific Research Foundation of Xidian University, and was part of 3DForMod project (ANR-17-EGAS-0002-01) funded in the frame of the JPI FACCE ERA-GAS call funded under European Union's Horizon 2020 research and innovation program (grant agreement No. 696356).

## Availability of data and materials

The datasets analysed during the current study are available from two hosts. The TLS data from Finland are owned by the FGI, and are freely available from the following repository ([https://helsinki-fi-my.sharepoint.com/:f/g/personal/jpyorala\\_ad\\_helsinki\\_fi/EvL7Zbq8VOpCqWMVIRIsWSIB06GH3HyOAAQ1rupwb3fOEXg?e=KdWmd7](https://helsinki-fi-my.sharepoint.com/:f/g/personal/jpyorala_ad_helsinki_fi/EvL7Zbq8VOpCqWMVIRIsWSIB06GH3HyOAAQ1rupwb3fOEXg?e=KdWmd7)). The Australia dataset is provided through the original publication (Calders et al. 2015), and can be freely accessed from the TERN AusCover data archive (<http://dx.doi.org/10.4227/05/542B766D5D00D>).

## Declarations

### Ethics approval and consent to participate

Not applicable.

### Consent for publication

Not applicable.

### Competing interests

The authors declare that they have no competing interests.

### Author details

<sup>1</sup>Department of Remote Sensing Science and Technology, School of Electronic Engineering, Xidian University, 710077 Xi'an, China. <sup>2</sup>State Key Laboratory of Information Engineering in Surveying Mapping and Remote Sensing, Wuhan University, 430079 Wuhan, China. <sup>3</sup>Department of Remote Sensing and Photogrammetry, Finnish Geospatial Research Institute, 02430 Masala, Finland. <sup>4</sup>Plant Systematics and Ecology Laboratory, Higher Teacher's Training College, University of Yaoundé I, BP 337 Yaoundé, Cameroon. <sup>5</sup>AMAP, Univ Montpellier, IRD, CNRS, CIRAD, INRAE, F-34000 Montpellier, France.

Received: 29 April 2021 Accepted: 10 August 2021

Published online: 10 October 2021

## References

- Åkerblom M, Raunonen P, Mäkipää R, Kaasalainen M (2017) Automatic tree species recognition with quantitative structure models. *Remote Sens Environ* 191:1–12. <https://doi.org/10.1016/j.rse.2016.12.002>
- Barbeito I, Dassot M, Bayer D, Collet C, Drössler L, Löf M, Del Rio M, Ruiz-Peinado R, Forrester DI, Bravo-Oviedo A, et al (2017) Terrestrial laser scanning reveals differences in crown structure of *fagus sylvatica* in mixed vs. pure european forests. *For Ecol Manag* 405:381–390

- Ben-Shabat Y, Avraham T, Lindenbaum M, Fischer A (2018) Graph based over-segmentation methods for 3d point clouds. *Comp Vision Image Underst* 174:12–23
- Burt A, Disney M, Calders K (2019) Extracting individual trees from lidar point clouds using treeSeg. *Methods Ecol Evol* 10(3):438–445
- Calders K, Adams J, Armston J, Bartholomeus H, Bauwens S, Bentley LP, Chave J, Danson FM, Demol M, Disney M, Gaulton R, Krishna Moororthy SM, Levick SR, Saarinen N, Schaaf C, Stovall A, Terry L, Wilkes P, Verbeeck H (2020) Terrestrial laser scanning in forest ecology: Expanding the horizon. *Remote Sens Environ* 251:112102. <https://doi.org/10.1016/j.rse.2020.112102>
- Calders K, Newnham G, Burt A, Murphy S, Raunonen P, Herold M, Culvenor D, Avitabile V, Disney M, Armston J, et al (2015) Nondestructive estimates of above-ground biomass using terrestrial laser scanning. *Methods Ecol Evol* 6(2):198–208
- Dijkstra EW (1959) A note on two problems in connexion with graphs. *Numer Math* 1(1):269–271
- Du S, Lindenbergh R, Ledoux H, Stoter J, Nan L (2019) Adtree: Accurate, detailed, and automatic modelling of laser-scanned trees. *Remote Sens* 11(18):2071. <http://doi.org/10.3390/rs11182074>
- Hackenberg J, Wassenberg M, Spiecker H, Sun D (2015) Non destructive method for biomass prediction combining tIs derived tree volume and wood density. *Forests* 6(4):1274–1300
- Heinzel J, Huber M (2018) Constrained spectral clustering of individual trees in dense forest using terrestrial laser scanning data. *Remote Sens* 10(7):1056
- Lau A, Bentley LP, Martius C, Shenkin A, Bartholomeus H, Raunonen P, Malhi Y, Jackson T, Herold M (2018) Quantifying branch architecture of tropical trees using terrestrial lidar and 3d modelling. *Trees* 32(5):1219–1231
- Liang X, Hyypää J, Kaartinen H, Lehtomäki M, Pyörälä J, Pfeifer N, Holopainen M, Brolly G, Francesco P, Hackenberg J, et al (2018) International benchmarking of terrestrial laser scanning approaches for forest inventories. *ISPRS J Photogramm Remote Sens* 144:137–179
- Liang X, Kankare V, Hyypää J, Wang Y, Kukko A, Haggrén H, Yu X, Kaartinen H, Jaakkola A, Guan F, et al (2016) Terrestrial laser scanning in forest inventories. *ISPRS J Photogramm Remote Sens* 115:63–77
- Liang X, Kankare V, Yu X, Hyypää J, Holopainen M (2014) Automated stem curve measurement using terrestrial laser scanning. *IEEE Trans Geosci Remote Sens* 52(3):1739–1748. <https://doi.org/10.1109/TGRS.2013.2253783>
- Liu J, Wang T, Skidmore AK, Jones S, Heurich M, Beudert B, Premier J (2019) Comparison of terrestrial lidar and digital hemispherical photography for estimating leaf angle distribution in european broadleaf beech forests. *ISPRS J Photogramm Remote Sens* 158:76–89
- Martin-Ducup O, Mofack GI, Wang D, Raunonen P, Ploton P, Sonké B, Barbier N, Coutron P, Pélissier R (2021) Evaluation of automated pipelines for tree and plot metrics estimation from tIs data in a tropical forest. *Ann Bot*. <https://doi.org/10.1093/aob/mcab051>
- Peng W, Zhang W, Liang X, Skidmore A, Yan G (2019) Quantification of occlusions influencing the tree stem curve retrieving from single-scan terrestrial laser scanning data. *For Ecosyst* 4:285–297. <https://doi.org/10.1186/s40663-019-0203-1>
- Pfeifer N, Höfle B, Briese C, Rutzinger M, Haring A (2008) Analysis of the backscattered energy in terrestrial laser scanning data. *Int Arch Photogramm Remote Sens Spat Inf Sci* 37:1045–1052
- Pueschel P, Newnham G, Rock G, Udelhoven T, Werner W, Hill J (2013) The influence of scan mode and circle fitting on tree stem detection, stem diameter and volume extraction from terrestrial laser scans. *ISPRS J Photogramm Remote Sens* 77:44–56. <https://doi.org/10.1016/j.isprsjprs.2012.12.001>
- Raunonen P, Casella E, Calders K, Murphy S, Åkerblom M, Kaasalainen M (2015) Massive-scale tree modelling from tIs data. *ISPRS Ann Photogramm Remote Sens Spat Inf Sci II-3/W4*:189–196
- Raunonen P, Kaasalainen M, Åkerblom M, Kaasalainen S, Kaartinen H, Vastaranta M, Holopainen M, Disney M, Lewis P (2013) Fast automatic precision tree models from terrestrial laser scanner data. *Remote Sens* 5(2):491–520
- Tao S, Wu F, Guo Q, Wang Y, Li W, Xue B, Hu X, Li P, Tian D, Li C, Yao H, Li Y, Xu G, Fang J (2015) Segmenting tree crowns from terrestrial and mobile lidar data by exploring ecological theories. *ISPRS J Photogramm Remote Sens* 110:66–76. <https://doi.org/10.1016/j.isprsjprs.2015.10.007>
- Trochta J, Krůček M, Vrška T, Král K (2017) 3d forest: An application for descriptions of three-dimensional forest structures using terrestrial lidar. *PLoS ONE* 12(5):1–17
- Wang D (2020) Unsupervised semantic and instance segmentation of forest point clouds. *ISPRS J Photogramm Remote Sens* 165:86–97. <https://doi.org/10.1016/j.isprsjprs.2020.04.020>
- Wang J, Chen X, Cao L, An F, Chen B, Xue L, Yun T (2019) Individual rubber tree segmentation based on ground-based lidar data and faster r-cnn of deep learning. *Forests* 10(9):793
- Wang D, Hollaus M, Puttonen E, Pfeifer N (2016) Automatic and self-adaptive stem reconstruction in landslide-affected forests. *Remote Sens* 8(12):974
- Wang D, Kankare V, Puttonen E, Hollaus M, Pfeifer N (2017) Reconstructing stem cross section shapes from terrestrial laser scanning. *IEEE Geosci Remote Sens Lett* 14(2):272–276
- Zhang W, Qi J, Wan P, Wang H, Xie D, Wang X, Yan G (2016) An easy-to-use airborne lidar data filtering method based on cloth simulation. *Remote Sens* 8(6):501
- Zhang W, Wan P, Wang T, Cai S, Chen Y, Jin X, Yan G (2019) A novel approach for the detection of standing tree stems from plot-level terrestrial laser scanning data. *Remote Sens* 11(2):211
- Zhong L, Cheng L, Xu H, Wu Y, Chen Y, Li M (2016) Segmentation of individual trees from tIs and mIs data. *IEEE J Sel Top Appl Earth Obs Remote Sens* 10(2):774–787

Submit your manuscript to a SpringerOpen® journal and benefit from:

- Convenient online submission
- Rigorous peer review
- Open access: articles freely available online
- High visibility within the field
- Retaining the copyright to your article

Submit your next manuscript at ► [springeropen.com](https://www.springeropen.com)