



Point-cloud segmentation of individual trees in complex natural forest scenes based on a trunk-growth method

Qianwei Liu^{1,2,3} · Weifeng Ma^{1,2,3} · Jianpeng Zhang^{1,2,3} ·
Yicheng Liu^{1,2,3} · Dongfan Xu⁴ · Jinliang Wang^{1,2,3}

Received: 11 December 2020 / Accepted: 13 January 2021 / Published online: 4 March 2021
© The Author(s) 2021

Abstract Forest resource management and ecological assessment have been recently supported by emerging technologies. Terrestrial laser scanning (TLS) is one that can be quickly and accurately used to obtain three-dimensional forest information, and create good representations of forest vertical structure. TLS data can be exploited for highly significant tasks, particularly the segmentation and information extraction for individual trees. However, the existing single-tree segmentation methods suffer from low segmentation accuracy and poor robustness, and hence do not lead to satisfactory results for natural forests in complex environments. In this paper, we propose a trunk-growth (TG) method for single-tree point-cloud segmentation, and apply this method to the natural forest scenes of Shangri-La City in Northwest Yunnan, China. First, the point normal vector and its Z-axis

component are used as trunk-growth constraints. Then, the points surrounding the trunk are searched to account for regrowth. Finally, the nearest distributed branch and leaf points are used to complete the individual tree segmentation. The results show that the TG method can effectively segment individual trees with an average *F*-score of 0.96. The proposed method applies to many types of trees with various growth shapes, and can effectively identify shrubs and herbs in complex scenes of natural forests. The promising outcomes of the TG method demonstrate the key advantages of combining plant morphology theory and LiDAR technology for advancing and optimizing forestry systems.

Keywords Terrestrial laser scanning · Point-cloud · Northwest Yunnan · Natural forests · Single-tree segmentation · Trunk-growth

Project funding: The work was supported by the National Natural Science Foundation of China (Grant Number 41961060), the Key Program of Basic Research of Yunnan Province, China (Grant Number 2019FA017), the Multi-government International Science and Technology Innovation Cooperation Key Project of National Key Research and Development Program of China (Grant Number 2018YFE0184300), the Program for Innovative Research Team in Science and Technology research and innovation fund (ysdyjs 2020058) in the University of Yunnan Province.

The online version is available at <https://www.springerlink.com>.

Corresponding editor: Tao Xu.

✉ Jinliang Wang
jliwang@ynnu.edu.cn

¹ Faculty of Geography, Yunnan Normal University, Kunming 650092, Yunnan, People's Republic of China

² Key Laboratory of Resources and Environmental Remote Sensing for Universities in Yunnan, Kunming 650500, Yunnan, People's Republic of China

Introduction

Effective monitoring and management of forest resources plays a vital role in maintaining sustainable forest development, monitoring and control of forest pests, protecting biodiversity, and assessing fire hazards (Oveland et al. 2017; Xiao et al. 2019; Ma et al. 2020; Windrim and Bryson 2020). Numerous studies have demonstrated that remote-sensing technologies can be used to effectively and reliably monitor

³ Center for Geospatial Information Engineering and Technology of Yunnan Province, Kunming 650500, Yunnan, People's Republic of China

⁴ Forestry College, Southwest Forestry University, Kunming 650224, Yunnan, People's Republic of China

and estimate forest resources (Strîmbu and Strîmbu 2015). As a popular remote-sensing technology, light detection and ranging (LiDAR) can be used to obtain information (such as target position, and height), characterize the spatial vegetation structure, and describe and monitor forest structure and functions (Liu et al. 2016, 2018, 2020; Lou et al. 2019; Yang et al. 2020).

The airborne laser scanning (ALS) method can be used to obtain a substantial amount of forest point-cloud data. However, the top-down scanning mode of this method only allows it to easily capture the crown part of a tree while largely overlooking its trunk and branch parts. This shortcoming limits ALS applicability, especially at the scale of individual trees (Zhong et al. 2017; Asadilla et al. 2020). Alternately, the terrestrial laser scanning (TLS) method adopts a bottom-up scanning mode which captures the trunk, branches, and leaf points with high density and integrity within a forest. This desirable TLS capability makes up for the afore-mentioned ALS shortcoming, and offers great advantages in the study of forest understory structures (Liang et al. 2012; Cabo et al. 2018; Ma et al. 2019; Wang et al. 2019). Reasonable and accurate segmentation of a single tree from TLS point-cloud data can be effectively utilized to obtain key information for any tree (Lu et al. 2020), such as its height (Unger et al. 2014), crown width (Hari-kumar et al. 2017), diameter-at-breast height (DBH) (Henning and Radtke 2006), and biomass (Lu et al. 2020). Such information is of great significance for quick and accurate understanding of forest resource information, and for eliminating the costs of traditional field surveys.

The TLS-based single-tree segmentation method is more challenging than the ALS-based one. Maas et al. (2008) first estimated the position and DBH of a single tree by detecting circles centered at the single-tree position, visualizing an imaginary cylinder within a certain range and segmenting the individual tree. Cabo et al. (2018) projected the cloud-point coordinates onto a two-dimensional plane and constructed a Voronoi diagram (Erwig 2000; Rusdhi et al. 2017) for single-tree segmentation based on the detected position. The above methods achieve good single-tree segmentation results in forests with sparse-dispersed trees. However, in natural forests affected by factors of sunlight and water, the stands are dense and individual trees are often of irregular shapes. Due to the lack of topological relationships among the TLS-based point-cloud data samples, canopies of individual trees with intersecting tree branches or overlapping canopies cannot be easily identified by cylinder or linear cutting methods. Xing et al. (2017) investigated the problem of extracting irregular-shaped trees of Mongolian oak (*Quercus mongolica* Fosch. ex Ledeb.) and achieved single-tree segmentation through point-cloud voxelization and hierarchical clustering. While this method works fairly well for artificial forests, i.e., plantations, individual tree segmentation

rate from 95 to 100%, its application in complex natural forests remains to be studied. Tao et al. (2015) proposed the comparative shortest-path (CSP) method. This method first uses the algorithm of density-based spatial clustering of applications with noise (DBSCAN) to identify individual trees. Then, a graph-theoretical method is followed to assign points to the shortest-path trunk. Finally, canopies are localized and single trees are thus segmented. However, shrubs and herbs are typically clustered in natural forests, and the DBSCAN-based identification of individual trees also causes serious over-segmentation.

To summarize, the existing single-tree segmentation methods are difficult to apply in natural forests with complex stand patterns. In this study, we seek to enhance the state of the art in single-tree segmentation through a trunk-growth (TG) algorithm and apply this to data collected in a natural forest in Shangri-La City, Northwest Yunnan, China. In this algorithm, key seed points are selected and used to extract the trunk of an individual tree. The branch points are then grown based on the trunk points. Finally, the nearest-neighbour distance allocates the branch and leaf points and completes the segmentation process. In this study, the *F*-score is used as an evaluation index to compare the TG method with various single-tree segmentation methods to study its segmentation accuracy and reliability.

Materials and methods

Study area

The study area is located in Shangri-La City (26° 52′–28° 52′ N, 99° 20′–100° 19′ E) of the Diqing Tibetan Autonomous Prefecture in Northwest Yunnan Province. The landform is essentially alpine with an average elevation of 3460 m a.s.l. The region is mountainous with a low temperature, monsoon climate. Forest cover is as high as 76% and the vegetation is rich with multiple species, (including *Pinus yunnanensis* Franch., *Pinus densata* Mast., *Picea asperata* Mast., and *Quercus aquifolioides* Rehd. et Wils.) (Fig. 1).

Data acquisition and pre-processing

Sample plot usually refers to a plot which fully reflects the average stand characteristics of the forest. Such a plot is typically a circle with a radius of 4–15 m (Liang et al. 2016). Two natural forest sample plots were selected in a natural forest in Shangri-La City in August 2016 and October 2019. A tape measure and laser rangefinder determined the extent of the plot, record the tree species and number of trees. Center coordinates, elevation, and slope were obtained with a handheld GPS instrument (Table 1). A Leica P40 TLS LiDAR was used to scan the plots and return the data in

Fig. 1 Maps of the study area: **a** China and the Northwest Yunnan Province; **b** location of the study area in Yunnan; **c** digital elevation model (DEM) of Shangri-La City

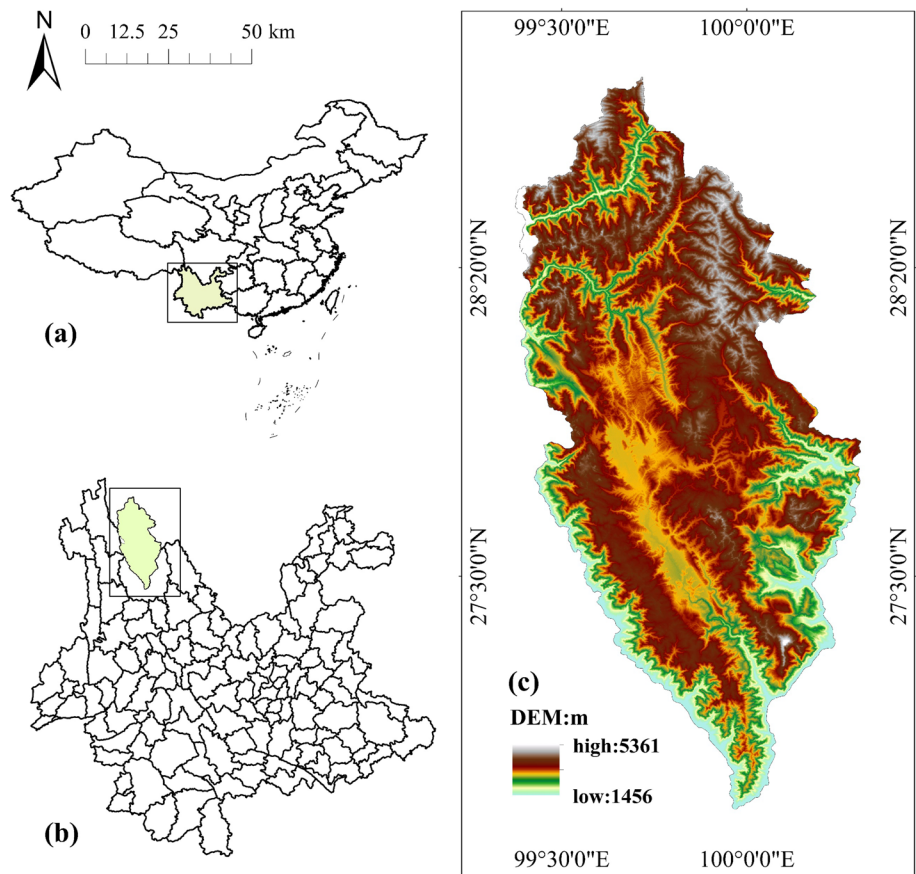


Table 1 Description of the environment of the two sample plots

Plot	Size (r) (m)	Center	Elevation (m)	Main species	General situation
1	10	N27° 38', E99° 45'	3269.07	<i>Pinus densata</i>	Herbs and shrubs are dense
2	15	N27° 48', E99° 59'	3580.00	<i>Picea asperata</i>	Herbs and shrubs are dense

Table 2 Specifications of the Leica P40 LiDAR

Indicator	Description
Range accuracy	1.2 mm + 10 ppm
3D position accuracy	3 mm @ 50 m and 6 mm @ 100 m
Wavelength	1550 nm (invisible); 658 nm (visible)
Scan rate	Up to 1,000,000 points per second
Field of view	360 (horizontal); 290 (vertical)
Range and reflectivity	Max range: 120 m (8%), 180 m (8%), 270 m (34%)
Range noise	Min range: 0.4 m 0.4 mm RMS @ 10 m 0.5 mm RMS @ 50 m

a multi-station mode (Table 2). The point-cloud data was spliced using Cyclone commercial software.

The two sample plots are sites with undulating topography where point-cloud data cannot be used directly and

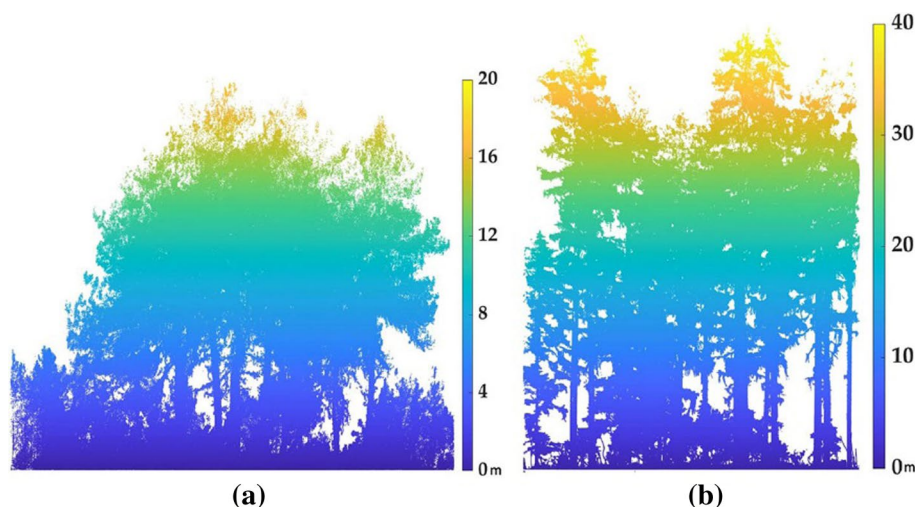
thus pre-processing steps are required. First, improved progressive triangulation filtering (Zhao et al. 2016) identifies ground points and carries out height normalization. The point-data is cropped to the size of the plot (plot 1 contains 3,638,235 points, plot 2 4,329,903 points, and stored in txt format) (Fig. 2).

Principle and steps of the trunk-growth algorithm

Principle

The trunk, branches, and leaves are closely related and largely inseparable; the trunk produces the branches and connects them with the leaves (Jin 2010). Based on plant morphology principles, the trunk of an individual tree is first extracted. Then the branches are grown out from the trunk. Finally, the nearest-neighbour method is used to distribute the branches and leaf points in order to complete the single-tree segmentation. Specifically, our proposed method

Fig. 2 Point-cloud data for the two forest plots: **a** *Pinus densata* point-cloud data for plot 1 after normalization; **b** *Picea asperata* point-cloud data for plot 2 after normalization



includes four steps of trunk extraction, fragment merging, trunk growth, and localization of branch and leaf points. Each of these four steps is discussed in detail below.

Trunk extraction

Extracting individual tree trunks is a prerequisite for single-tree identification and segmentation in natural forests. Liang et al. (2012) calculated the normal vector at each cloud point and took the Z-axis component Z_n (0–1) of the normal vector as the characteristic value of that point on the vertical plane. The smaller that value is, the closer the point to the vertical plane. This observation provides guidance for the extraction of the trunk of an individual tree. Calculating the normal vector at a certain point can be formulated as the problem of fitting a plane to that point and its k -neighbouring points. Assuming that P_i is one point in the collected point-cloud data, the covariance matrix based on P_i and its k -neighbouring points is given by (Guo et al. 2018; Zhang et al. 2019):

$$C(P_i) = \sum_{i=1}^k \left(P_i - \frac{1}{k} \sum_{i=1}^k P_i \right) \left(P_i - \frac{1}{k} \sum_{i=1}^k P_i \right)^T = \sum_{j=1}^3 e_j \lambda_j e_j^T \quad (1)$$

In (1), $j = \{1, 2, 3\}$ represents the set of eigenvalues and corresponding eigenvectors, that is, λ_j is the j th eigenvalue ($\lambda_1 \leq \lambda_2 \leq \lambda_3$) with a corresponding eigenvector e_j . The eigenvector (e_1) associated with the smallest eigenvalue is the point normal vector. As $n = [0, 0, 1]$, Z_n can be obtained as the magnitude of the cross product of e_1 and n (Liang et al. 2012):

$$Z_n = |e_1 \times n| \quad (2)$$

Taking the Z_n value and the normal vector as trunk extraction constraints, the region-growth algorithm is enforced to extract the trunk of each tree. The pseudo-code of this enhanced trunk extraction algorithm is as follows:

ALGORITHM 1 - TRUNK EXTRACTION

Input: Point clouds P and k , Normal Vector (NV) and Z_n

```

class = 0, cluster = 0,
for every point  $P_i \in P$  do
if  $Z_n < 0.05$  and  $(P_i(0,3) > 0.05$  and  $P_i(0,3) < 0.20)$  in seed point (SP)
class = class + 1, cluster = class
while SP is not empty
calculate the NV angle between the SP and the adjacent point
if angle < threshold 1, adjacent point class = cluster
calculate the  $Z_n$  difference between the SP and the adjacent point
if the difference < threshold 2, adjacent point in SP
end for

```

Output cluster

The algorithm traverses the point-cloud data, sets the points with Z_n values below 0.05 as seed points, and then searches and traverses the adjacent points. When the normal vector angle between the seed point and a neighbour point is below a certain threshold, the neighbour point is assigned to the same trunk region. If the Z_n value difference between the seed and neighbour points is small, the neighbour point is set as a seed point for further growth. While existing area growth methods require manual intervention, our algorithm can find seed points for trunk extraction in a highly automated manner.

Trunk growth

The branches connect and bridge the trunk with the leaves. The branches are assumed to be connected with the trunk, and the point-cloud data satisfy this assumption. By finding the adjacent points around the trunk, the trunk can be grown again. The pseudo-code of the trunk-growth algorithm is as follows:

ALGORITHM 2–TRUNK GROWTH

Input: Point clouds P and k , **Normal Vector (NV)** and Z_n

```

for every point  $P_i \in P$  do
  count = 0
  while  $k$  adjacent point
    if trunk class known and only, count = count + 1
    else continue
  if count >  $k/2$  in seed point (SP)
    while seed point is not empty
      calculate the NV angle between the SP and the adjacent point
      if angle > threshold 1, adjacent point in trunk
        calculate the distance between SP and adjacent point
        if distance < threshold 3, adjacent point in SP
    end for

```

Fragment merging

In first step, automating the search for seed points may result in the growth of some non-trunk points. These points produce many categories that are not trunk and should not be merged in categories. The points can only meet the growth conditions in a small area, resulting in a low number of category points. Therefore, in step (1), all category points are counted, and the categories with fewer points are removed. This completes the trunk extraction step.

This trunk growth algorithm is similar to the trunk extraction algorithm in step (1). The former modifies the selection conditions of seed points in order to enforce branch growth. The nearest-neighbour point of the trunk is taken as the seed point and calculates the normal-vector angle between the seed and k -nearest points. When certain conditions are satisfied, the adjacent point is assigned the same category as the seed one. When the distance between the adjacent and the seed points does not exceed a certain threshold value, the two points are considered to belong to the same branch trunk. The threshold represents the maximum length that a branch can grow and is usually determined by the growth of the tree.

Distribution of branches and leaves

The above steps result in the extraction of the trunk and a part of the branches of an individual tree. In this current step, the remaining branches and leaves are added. In fact, the structure of natural forests is often associated with vertical stratification. The strata are classified according to the different heights to which their plants grow: a herb layer, a shrub layer, and a tree layer (Shugart et al. 2010; Zhao et al. 2013). The United Nations Food and Agriculture Organization (FAO) defines understory vegetation as perennial woody plants with heights > 0.5 and < 5.0 m at maturity (FAO 2003). Therefore, a height between 0.5 and 5.0 m was selected as the vertical stratification boundary to distinguish the tree and understory vegetation layers. A value of -1 was assigned to the category below this boundary, the distance from the branch point to the surrounding known points was calculated, and the nearest category for assignment was selected until all points were classified. The single-tree segmentation steps are therefore completed.

Segmentation performance evaluation

When an individual tree is successfully segmented, the result is counted as a true positive (TP). Otherwise, when a tree is incorrectly segmented as another nearby tree, the result is counted as a false negative (FN). When no individual tree exists but data points are segmented as belonging to one, the result is counted as a false positive (FP). Based on these

values, we define the recall (R), the precision (P), and the F -score, respectively, as follows:

$$R = \frac{TP}{TP + FN} \quad (3)$$

$$P = \frac{TP}{TP + FP} \quad (4)$$

$$F = 2 \times \frac{R \times P}{R + P} \quad (5)$$

These metrics are computed and used for evaluating the single-tree segmentation outcomes. In addition, the performance of our method was compared against that of the comparative shortest-path (CSP) method (Tao et al. 2015). First of all, CSP uses DBSCAN clustering for trunk identification, calculates the DBH, and connects all point clouds to form a cloud map. Then, the transportation path from each point to the adjacent trunk diameter is calculated. Finally, tree crown points are assigned to the shortest transportation path, which is calculated as follows:

$$D_{P \rightarrow Trunk}^N = \frac{D_{p \rightarrow Trunk}}{DBH^{\frac{2}{3}}} \quad (6)$$

where $D_{p \rightarrow Trunk}$ is the transportation distance from the point p to the trunk DBH. Further details of the comparative shortest-path (CSP) algorithm are not discussed herein, but

Fig. 3 Comparison of single-tree segmentation results of the TG, CSP, and PCS algorithms

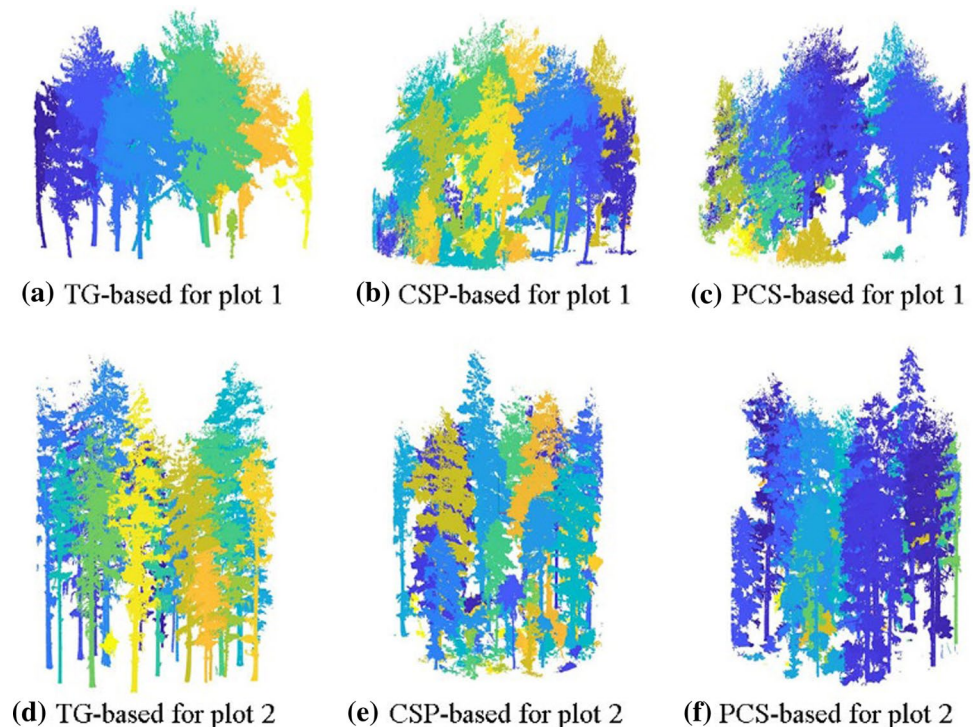


Fig. 4 Comparison of the shrub-and-herb map segmentation results (after separation from the tree layer) of the TG, CSP, and PCS algorithms

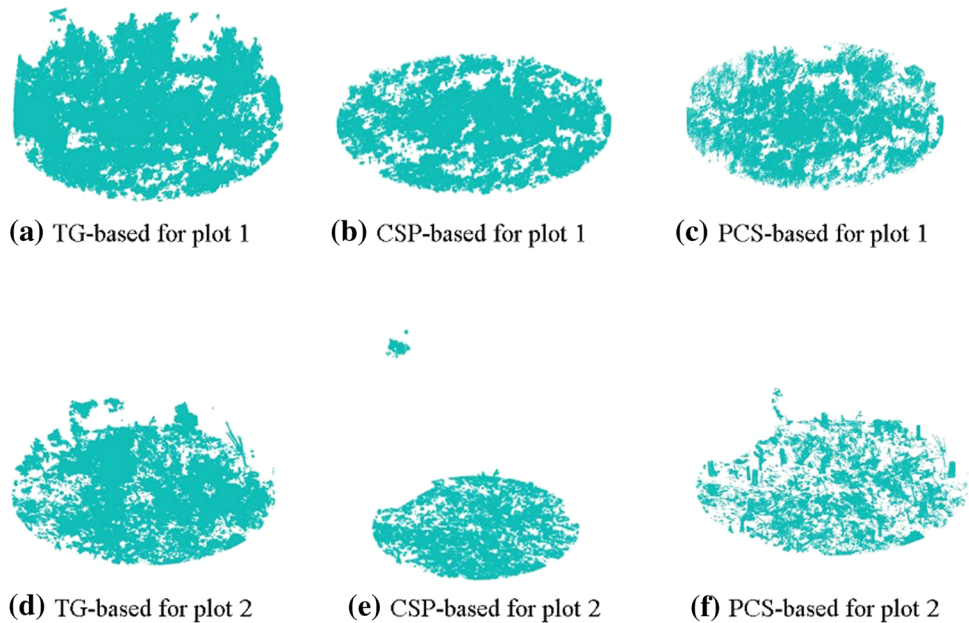


Table 3 Performance measures for three single-tree segmentation methods applied to two plots

Method	Plot	Number of trees	Number of segmented trees	TP	FP	FN	R	P	F
TG	1	13	14	13	1	0	1.00	0.93	0.96
CSP	1	13	60	13	46	1	1.00	0.22	0.36
PCS	1	13	36	11	25	2	0.85	0.31	0.45
TG	2	21	21	20	1	1	0.95	0.95	0.95
CSP	2	21	70	21	49	0	1.00	0.30	0.46
PCS	2	21	51	14	37	7	0.67	0.27	0.38

the interested reader might consult Liu and Ramakrishnan (2001) or Zhang et al. (2020).

Results

Li et al. (2012a) proposed an ALS-based point-cloud segmentation (PCS) method that combines regional growth with thresholding. In this method, the tree height is set as that of the highest point in the point-cloud data, and a tree is obtained through regional growth. Then, the distances between a known individual tree and other points are computed for step-by-step single-tree segmentation. This algorithm is suitable for single-tree segmentation based on ALS data or, to some extent, TLS data. In our study, we compared the performance outcomes obtained by our TG algorithm against those of the classical PCS and CSP algorithms. The comparative results are shown in Figs. 3 and 4 (the segmented trees are displayed in different colours), and the final accuracy results are shown in Table 3. Additionally, we used software LiDAR 360 to implement CSP and PCS algorithm

in laptop (Windows 10, CPU-i5-4210H@2.9 GHz, Memory 16 g) and Matlab 2019b to run TG. The runtime of TG algorithm was 7 min and 32 s in plot 1, and 7 min and 56 s in plot 2.

The TG method performed well on both plots (Table 3). For plot 1, the R value reached 1.00, and each tree was well-segmented, with an F -score of 0.96. For plot 2, the R value was 0.95 and the F -score 0.95, but one of the trees grew so poorly that it was wrongly assigned to the shrub layer during fragment merging. The F -score of the TG method is the highest among the three methods. Although the CSP method can segment individual trees fairly well, it is prone to errors due to the influence of shrubs and herbaceous plants. These errors result in a lower P value, which ultimately decreases the F -score. The PCS method showed serious under-segmentation problems for both plots. Also, the PCS method lacks constraints and cannot easily separate two individual trees with intersecting canopies. This, in turn, leads to a low R value and poor single-tree segmentation performance. Nevertheless, the TG and CSP methods have

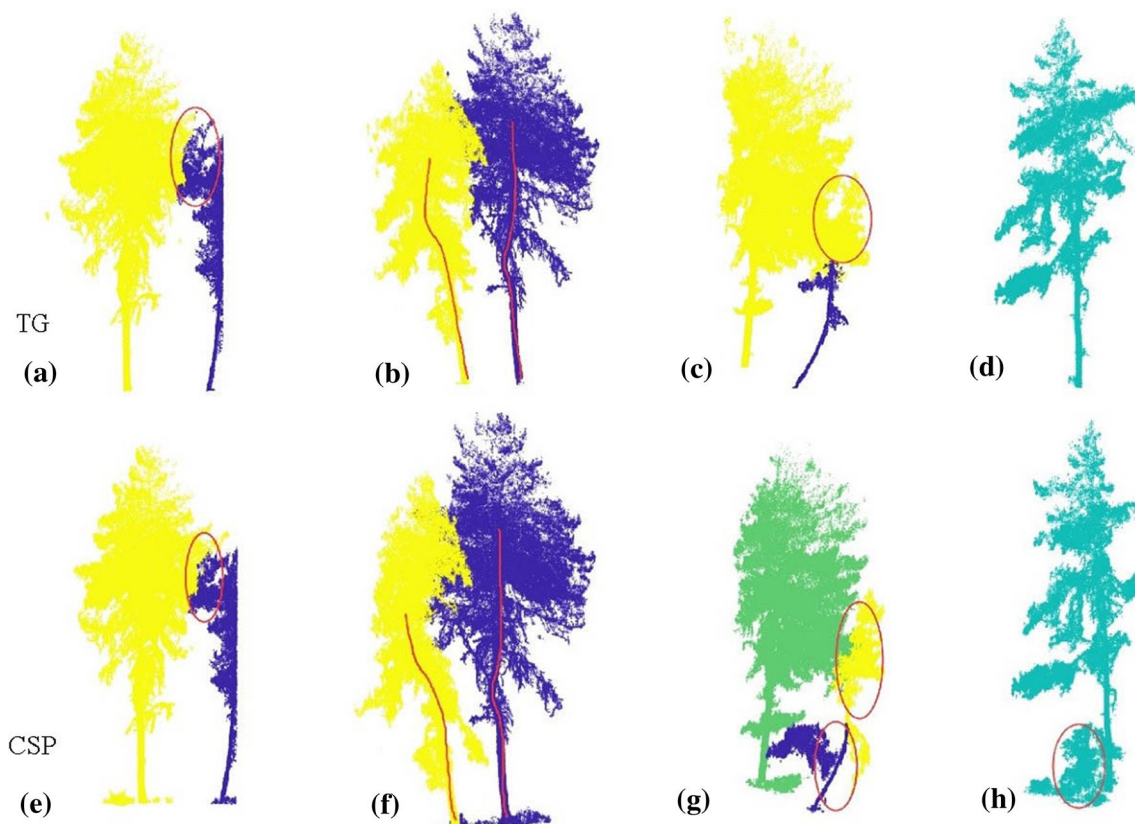


Fig. 5 Comparison of the segmentation results for different tree shapes; **a–d** TG-based segmentation results; **e–h** CSP-based segmentation results; **a, e** canopy segmentation for standing trees; **b, f**

canopy segmentation for irregular individual trees; **c, g** canopy segmentation for inclined individual trees; **d, h** separation of shrubs and herbs from trees

prior constraints for identifying individual trees, and hence they can be used to segment individuals more accurately.

Discussion

Comparison of the TG and CSP segmentation results with different tree shapes

The TG and CSP methods (which exhibited higher R values than the PCS method) were compared on tasks of canopy segmentation and segmentation of individual trees with irregular trunk shapes.

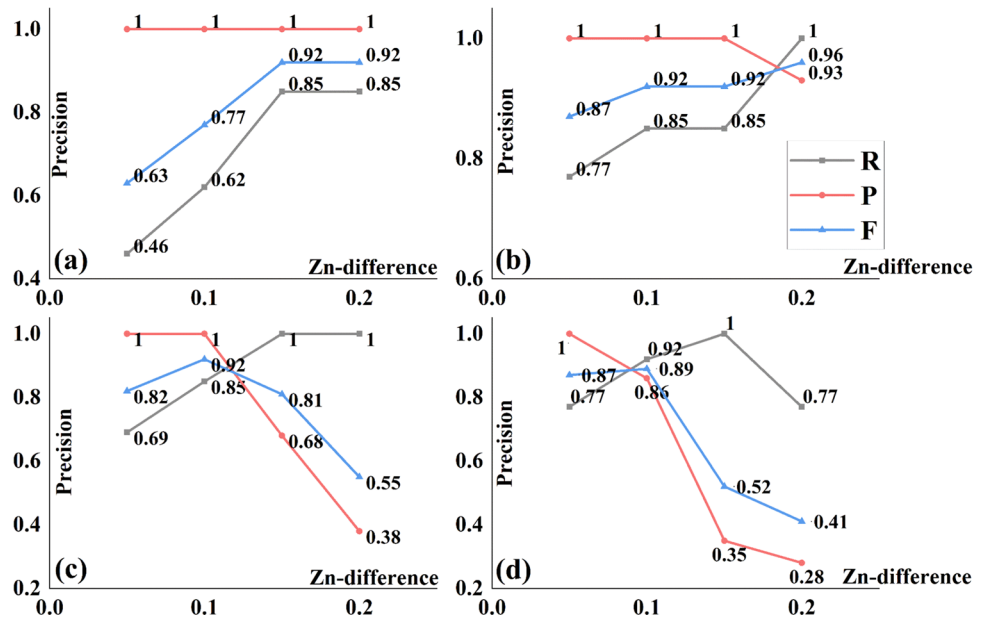
(1) *Canopy segmentation*: Both the TG and CSP methods resulted in canopy over-segmentation (Fig. 5a, e). Naturally formed individual trees in natural forests usually have no restrictions on their growth locations. The distance between two individual trees is also typically uneven, causing crowns to intersect during growth. The points of the two trees blend. At present, high-quality canopy segmentation cannot be obtained using either

the TG or the CSP methods. Solutions for this problem should be investigated in future work.

(2) *Irregular trunk and canopy segmentation*: Individuals of the same species in a stand compete for nutrients due to sunlight, moisture, and other factors. In fact, the more nutrients a single tree obtains, the more divergent the growth of its branches and leaves becomes. With limited space, trees with few nutrients suffer more, and exhibit slow growth and poor development. Likewise, the conventional growth direction of a tree species may be changed to a certain extent. For example, the trunk may take an S shape (Fig. 5b, e). Both the TG and CSP methods handle better the identification and segmentation of such irregular-shaped trees. In particular, the TG method assigns crown points according to the nearest tree trunks, and results in good performance under appropriate parameter settings. The CSP method identifies the crown points by seeking the shortest path within a graph representation.

(3) *Segmentation of trees with inclined trunks* (Fig. 5c, g): For an individual tree, the TG method identifies the trunk, and then continues to grow the trunk and branch points. This approach enables the identification of individuals with large tilt angles, while canopy segmentation may

Fig. 6 Effects of the normal vector angle (threshold 1) and the Z_n value difference (threshold 2) on the recall (R), precision (P), and F -score (F) of the TG-based single-tree segmentation outcomes; **a** the normal vector angle is 10° ; **b** the normal vector angle is 20° ; **c** the normal vector angle is 30° ; **d** the normal vector angle is 40°



be unsatisfactory. The CSP method performs poorly on inclined trunks due to its tendency to split a tree into two segments. This limitation can be ascribed to the fact that the CSP algorithm prioritizes the identification of a single tree at DBH. When a tree has a large tilt angle or forms a

fallen tree with a trunk height below 1.3 m, the tree will be wrongly identified or missed altogether. For handling such a tree, the TG algorithm has an obvious advantage. (4) *Identification of shrubs and herbs*: Shrubs and herbaceous plants are competitors of trees in natural forests.

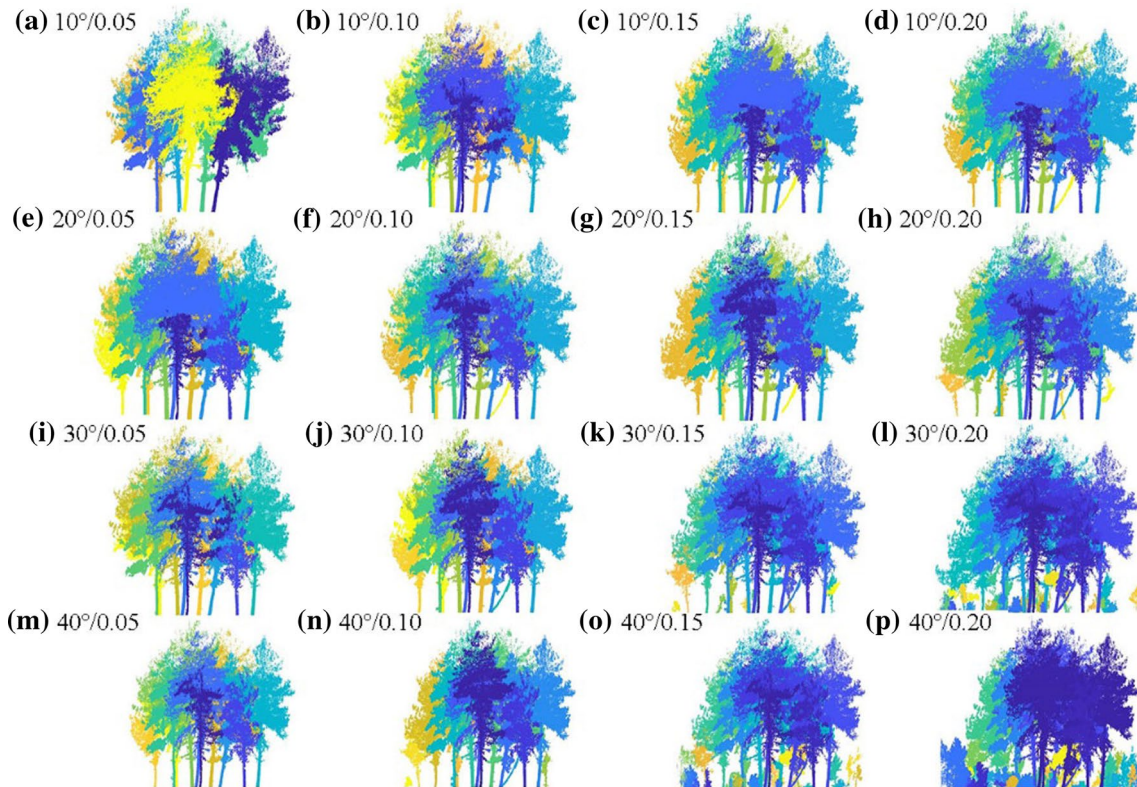


Fig. 7 Visual segmentation results with different threshold combinations (threshold 1/threshold 2)

Studies have demonstrated that shrubs and herbs usually interfere with the estimation of single-tree parameters (tree position, DBH, etc.) (Li et al. 2012b). As shown in Fig. 5d, h, the CSP-based single-tree segmentation cannot accurately separate the trunks, herbaceous vegetation, and shrubs. This is because the algorithm considers all points in a sample plot to be tree points, ignoring the complexity of natural forests. Indeed, separating non-trunk points near the trunk using the shortest-path method is challenging, and both types of points are often mixed into one class. The TG algorithm sets a fixed height that can effectively separate the understory environment (including shrubs and herbs) from the trees.

Parameter sensitivity for the TG method

In the proposed TG method, the trunk extraction process obviously has a great impact on the final crown segmentation. Trunk extraction is highly influenced in turn by the selection of two growth parameters: the normal vector angle (threshold 1) and the Z_n value difference (threshold 2) between the seed and neighbour points. In our work, we vary these parameters in the intervals $[10^\circ, 40^\circ]$ and $[0.05, 0.20]$, respectively, and investigate the parameter effects on the recall, precision, and F -score of the single-tree segmentation outcomes (Fig. 6).

When threshold 1 is 10° and threshold 2 is low, the results of trunk extraction and subsequent growth are poor, with an R score of only 0.46. In fact, many individual trees that do not meet the conditions are wrongly included in the shrub and herb layers. By increasing threshold 2, the segmentation accuracy also increases, reaching the peak at 0.15 after which the accuracy does not increase. When threshold 1 is 20° , and threshold 2 is between 0.05 and 0.15, the P value is the highest. When threshold 2 is 0.20, the P value decreases, while R reaches the highest value of 1.00. In addition, all individual trees are segmented at that threshold, but over-segmentation inevitably occurs and some of the shrubs and herbs are wrongly counted as tree parts. When threshold 1 is 30° , R reaches the highest value of 1.00 at a threshold 2 value of 0.15, and all individual trees are segmented. As threshold 2 increases, more and more non-tree parts are wrongly segmented, and P gradually decreases. When threshold 1 is 40° and threshold 2 is 0.15, all individual trees are segmented, but over-segmentation occurs as well. When threshold 2 reaches 0.20, multiple trees are merged together, and the F -score and R show trends of deterioration. When R reaches 1, the F -score curve gets closer to the P curve. When threshold 1 is unchanged, the TG-based single-tree segmentation accuracy increases with the increase of threshold 2, then decreases, showing a transition from under-segmentation to over-segmentation (Fig. 7).

This demonstrated sensitivity of the TG algorithm to the two thresholds is the key to performance tuning in single-tree segmentation.

Recommended TG parameter settings

In this paper, the proposed method is based on the patterns of plant morphology. The characteristic Z_n value is used as an essential constraint for trunk extraction, branch growth, and separation of understory vegetation. Good results were obtained in the two study plots. The calculation of Z_n requires k neighbour points. In earlier studies, the value of k was usually taken as between 8 and 32 (Zhang et al. 2018). Fewer neighbourhood points are expected to lead to incomplete segmentation. However, more points increase the computational cost, so in the TG method, we take the median value of 20 as the number of domain points. For plot 1, the threshold 1 value is 30° , the threshold 2 value is 0.14, and a height of 3 m is the threshold between trees and understory vegetation. For plot 2, the threshold 1 value is 40° , threshold 2 is 0.10, and the height threshold between trees and understory vegetation is 5 m. Therefore, we recommend the following parameter ranges: a threshold 1 range of 30° – 40° , a threshold 2 range of 0.1–0.2, and an understory height threshold of 3.0–5.0 m.

When the trunk of a forest tree is blocked and has no associated point-cloud data, the branch points cannot be grown; the canopy points are better assigned to the closest category and the Z_n value becomes more sensitive in upright trees. Also, trees with bending at the roots cannot be easily segmented by the TG method.

Botany is the science of studying plant morphology, ecology, distribution, etc. It provides a solid foundation for the study of LiDAR-based forestry methods. Tao et al. (2015) proposed the CSP algorithm based on plant ecology. Burt et al. (2018) carried out tree segmentation using the volume of the crown, the height, and the allometric growth relationship of crown width. Our paper summarizes the shortcomings of previous studies and proposes the TG algorithm according to the idea of “the trunk grows the branches, and the branches connect the leaves”. To a certain extent, the TG algorithm can be better applied to the sample plots in the natural stand in Shangri-La City. For future work, an in-depth study would be made to combine plant morphology theory with point-cloud data analysis.

Conclusions

Effective and accurate approaches for single-tree segmentation help with quickly understanding single-tree information and achieving efficient management of forest resources. Such approaches are highly needed for monitoring forest

biodiversity, water, and soil conservation capabilities. Based on the plant morphology theory, we sought to propose a novel method named the trunk-growth (TG) method. In this work, the segmentation accuracy and applicability of our algorithm were tested on point-cloud data collected for forest sample plots in a natural forest in Shangri-La City, northwest Yunnan, China. The performance of the TG method was evaluated using the *F*-score, comparing it with the CSP and PCS methods. The *F*-scores of CSP and PCS were in the range of 0.36–0.46, while the average TG *F*-score was 0.96, and the segmentation accuracy of TG was more successful. The trunk-growth method is able to handle inclined trees and more effectively reduce the interference of understory vegetation than the other methods.

Nevertheless, the TG algorithm expands segments outward by extracting the single-tree trunk as the core because such a trunk is usually limited by neighbouring trunks. When the trunk is blocked in the canopy, some of the required data becomes missing, posing difficulties on TG-based canopy segmentation. The TG parameters need to be adjusted continuously to achieve the best results. Besides, the Z_n value was used as the basis for extracting the trunk and can also be replaced by curvature or other feature values when the trunk is difficult to extract. Further performance improvements in canopy segmentation are expected to be the focus of future research.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

- Asadilla YSP, Ümüt HLK, Abdulla A, Maierdang KYM (2020) Terrestrial laser scanning for retrieving the structural parameters of *Populus euphratica* riparian forests in the lower reaches of the Tarim River, NW China. *Acta Ecol Sin* 40:4555–4565
- Burt A, Disney M, Calders K (2018) Extracting individual trees from LiDAR point clouds using tree-seg. *Methods Ecol Evol* 10:1–8
- Cabo C, Ordóñez C, López-Sánchez CA, Armesto J (2018) Automatic dendrometry: tree detection, tree height and diameter estimation using terrestrial laser scanning. *Int J Appl Earth Obs Geoinf* 69:164–174
- Erwig M (2000) The graph Voronoi diagram with applications. *Netw: Int J* 36(3):156–163
- FAO (Food and Agriculture Organization of the United Nations) (2003). Global forest resources assessment update: terms and definitions (final version). Forest Resources Assessment Programme, Rome, pp 1–34
- Guo QH, Su Y, Hu T, Liu J (2018) LiDAR principles, processing and applications in forest ecology. Higher Education Press, Beijing, pp 111–131
- Harikumar A, Bovolo F, Bruzzone L (2017) An internal crown geometric model for conifer species classification with high-density LiDAR data. *IEEE Trans Geosci Remote Sens* 55:2924–2940
- Henning JG, Radtke PJ (2006) Detailed stem measurements of standing trees from ground-based scanning LiDAR. *For Sci* 52:67–80
- Jin YG (2010) Botany, 2nd edn. Science Press, Beijing, pp 124–149
- Li D, Pang Y, Yue CR, Zhao D, Xu GC (2012a) Extraction of individual tree DBH and height based on terrestrial laser scanner data. *J Beijing For Univ* 34:79–86 ((in Chinese))
- Li WK, Guo QH, Jakubowski MK, Kelly M (2012b) A new method for segmenting individual trees from the LiDAR point cloud. *Photogrammetric Engineering and Remote Sensing* 78:75–84
- Liang XL, Kankare V, Hyypä J, Wang YS, Kukko A, Haggren H, Yu XW, Kaartinen H, Jaakkola A, Guan FY, Holopainen M, Vastaranta M (2016) Terrestrial laser scanning in forest inventories. *ISPRS J Photogramm Remote Sens* 115:63–77
- Liang XL, Litkey P, Hyypä J, Kaartinen H, Vastaranta M, Holopainen M (2012) Automatic stem mapping using single-scan terrestrial laser scanning. *IEEE Trans Geosci Remote Sens* 50:661–670
- Liu GLG, Ramakrishnan KG (2001) A*Prune: an algorithm for finding K shortest paths subject to multiple constraints. In: Proceedings IEEE INFOCOM 2001. Conference on computer communications. 20th annual joint conference of the IEEE computer and communications society, (Cat. No. 01CH37213), vol 2, pp 743–749
- Liu LX, Pang Y, Li ZY (2016) Individual tree DBH and height estimation using terrestrial laser scanning (TLS) in a subtropical Forest. *Sci Silvae Sin* 52:26–37 ((in Chinese))
- Liu XL, Sui LC, Bai YF, Zhao D, Zhao YJ, Liu YS, Zhai QP (2020) Biomass estimation of *Caragana microphylla* in the shrub-encroached grassland based on terrestrial laser scanning. *J Remote Sens* 24:894–903 ((in Chinese))
- Liu GJ, Wang JL, Dong PL, Chen Y, Liu ZY (2018) Estimating individual tree height and diameter at breast height (DBH) from terrestrial laser scanning (TLS) data at plot level. *Forests* 9:398–417
- Lou YB, Huang HY, Tang LY, Chen CC, Zhang H (2019) Tree height and diameter extraction with 3D reconstruction in a forest based on TLS. *Remote Sens Technol Appl* 34:243–252 ((in Chinese))
- Lu JB, Wang H, Qin SH, Cao L, Pu RL, Li GL, Sun J (2020) Estimation of aboveground biomass of *Robinia pseudoacacia* forest in the Yellow River delta based on UAV and Backpack LiDAR point clouds. *Int J Appl Earth Obs Geoinf* 86:102014
- Ma ZY, Pang Y, Li ZY, Lu H, Liu LX, Chen BW (2019) Fine classification of near-ground point cloud based on terrestrial laser scanning and detection of forest fallen wood. *J Remote Sens* 23:743–755 ((in Chinese))
- Ma ZY, Pang Y, Wang D, Liang XJ, Chen BW, Lu H, Weinacker H, Koch B (2020) Individual tree crown segmentation of a larch plantation using airborne laser scanning data based on region growing and canopy morphology features. *Remote Sens* 12:1078
- Maas HG, Bienert A, Scheller S, Keane E (2008) Automatic forest inventory parameter determination from terrestrial laser scanner data. *Int J Remote Sens* 29:1366–5901
- Oveland I, Hauglin M, Gobakken T, Næsset E, Maalen-Johansen I (2017) Automatic estimation of tree position and stem diameter using a moving terrestrial laser scanner. *Remote Sens* 9:350
- Rushdi AA, Swiler LP, Phipps ET, D'Elia M, Ebeida MS (2017) VPS: Voronoi piecewise surrogate models for high-dimensional data fitting. *Int J Uncertain Quantif* 7(1):1–21

- Shugart HH, Saatchi S, Hall FG (2010) Importance of structure and its measurement in quantifying function of forest ecosystems. *J Geophys Res* 115(G2):1–16
- Strîmbu VF, Strîmbu BM (2015) A graph-based segmentation algorithm for tree crown extraction using airborne LiDAR data. *ISPRS J Photogramm Remote Sens* 104:30–43
- Tao SL, Wu FF, Guo QH, Wang YC, Li WK, Xue BL, Hu XY, Li P, Tian D, Li C, Yao H, Li YM, Xu GC, Fang JY (2015) Segmenting tree crowns from terrestrial and mobile LiDAR data by exploring ecological theories. *ISPRS J Photogramm Remote Sens* 110:66–76
- Unger DR, Hung IK, Brooks R, Williams H (2014) Estimating number of trees, tree height and crown width using LiDAR data. *Gisci Remote Sens* 51:227–238
- Wang JM, Chen XX, Cao L, An F, Chen BQ, Xue LF, Yun T (2019) Individual rubber tree segmentation based on ground-based LiDAR data and faster R-CNN of deep learning. *Forests* 10:793–813
- Windrim L, Bryson M (2020) Detection, segmentation, and model fitting of individual tree stems from airborne laser scanning of forests using deep learning. *Remote Sens* 12:1469
- Xiao W, Zaforemska A, Smigaj M, Wang YS, Gaulton R (2019) Mean shift segmentation assessment for individual forest tree delineation from airborne LiDAR data. *Remote Sens* 11:1263
- Xing WL, Xing YQ, Huang Y, Qu L, You HT (2017) Individual tree segmentation of TLS point cloud data based on clustering of voxels layer by layer. *J Cent South Univ For Technol* 37:58–6471 ((in Chinese))
- Yang QY, Dong ZY, Ma ZY, Wu Y, Cui Q, Lu H (2020) Coniferous forest crown segmentation algorithm of UAV images based on SFM. *Trans Chin Soc Agric Mach* 51:181–190 ((in Chinese))
- Zhang JP, Wang JL, Liu GJ (2020) Vertical structure classification of a forest sample plot based on point cloud data. *J Indian Soc Remote Sens* 48(8):1215–1222
- Zhang WM, Wan P, Wang TJ, Cai SS, Chen YM, Jin XL, Yan GJ (2019) A novel approach for the detection of standing tree stems from plot-level terrestrial laser scanning data. *Remote Sens* 11:211
- Zhang WM, Wu X, Gao YK, Li HB (2018) A simplification algorithm for point cloud based on extraction. *Opt Tech* 44:733–735
- Zhao J, Li J, Liu QH (2013) Review of forest vertical structure parameter inversion based on remote sensing technology. *J Remote Sens* 17:697–716 ((in Chinese))
- Zhao XQ, Guo QH, Su YJ, Xue BL (2016) Improved progressive TIN densification filtering algorithm for airborne LiDAR data in forested areas. *ISPRS J Photogramm Remote Sens* 117:79–91
- Zhong LS, Cheng L, Xu H, Wu Y, Chen YM, Li MC (2017) Segmentation of individual trees from TLS and MLS data. *IEEE J Sel Top Appl Earth Obs Remote Sens* 10:774–787

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.