



Article

Point Cloud Inversion: A Novel Approach for the Localization of Trees in Forests from TLS Data

Shaobo Xia ^{1,2} , Dong Chen ³ , Jiju Peethambaran ⁴ , Pu Wang ⁵ and Sheng Xu ^{2,*}

¹ College of Environment and Resources, Zhejiang A&F University, Hangzhou 311300, China; shaobo.xia@zafu.edu.cn

² College of Information Science and Technology, Nanjing Forestry University, Nanjing 210037, China

³ College of Civil Engineering, Nanjing Forestry University, Nanjing 210037, China; chendong@njfu.edu.cn

⁴ Department of Math & Computing Science, Saint Marys University, Halifax, NS B3P 2M6, Canada; jiju.poovvancheri@smu.ca

⁵ Key Laboratory of Digital Earth Science, Aerospace Information Research Institute, Chinese Academy of Sciences, Beijing 100094, China; wangpu18@mails.ucas.ac.cn

* Correspondence: xusheng@njfu.edu.cn

Abstract: Tree localization in point clouds of forest scenes is critical in the forest inventory. Most of the existing methods proposed for TLS forest data are based on model fitting or point-wise features which are time-consuming, sensitive to data incompleteness and complex tree structures. Furthermore, these methods often require lots of preprocessing such as ground filtering and noise removal. The fast and easy-to-use top-based methods that are widely applied in processing ALS point clouds are not applicable in localizing trees in TLS point clouds due to the data incompleteness and complex canopy structures. The objective of this study is to make the top-based methods applicable to TLS forest point clouds. To this end, a novel point cloud transformation is presented, which enhances the visual salience of tree instances and makes the top-based methods adapting to TLS forest scenes. The input for the proposed method is the raw point clouds and no other pre-processing steps are needed. The new method is tested on an international benchmark and the experimental results demonstrate its necessity and effectiveness. Finally, the proposed method has the potential to benefit other object localization tasks in different scenes based on detailed analysis and tests.

Keywords: tree localization; tree instances; point cloud transformation; local maximums



Citation: Xia, S.; Chen, D.; Peethambaran, J.; Wang, P.; Xu, S. Point Cloud Inversion: A Novel Approach for the Localization of Trees in Forests from TLS Data. *Remote Sens.* **2021**, *13*, 338. <https://doi.org/10.3390/rs13030338>

Academic Editor: Mohammad Awrangjeb
Received: 5 December 2020
Accepted: 18 January 2021
Published: 20 January 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Terrestrial Laser Scanning (TLS) has become popular in plot-level tree mapping due to its ability in penetrating canopies and acquiring 3D fine-grained structures of vegetation [1,2]. As tree instances form the basis of many applications such as stem mapping [3,4], tree volume and diameter at breast height (DBH) measurement [5,6], reconstruction [7,8], data registration [9], and biophysical parameter retrieval (e.g., leaf area index) [10], it is necessary to extract tree instances from point clouds in the preprocessing. In this letter, we focus on the tree localization which is a sub-step in the tree instance mapping, aiming at localizing trees in point clouds.

Most of the previous tree localization methods for TLS point clouds can be categorized into two classes: model-fitting methods and feature-based methods. The model-fitting methods treat the tree localization as the detection of stems from point clouds [11–13]. Maas et al. [14] divide tree point clouds into slices along the vertical direction and the circle fitting is applied to the sliced points, then model inliers of fitted circles will be recognized as stems. Similarly, Liang et al. [6] use a cylinder fitting strategy to find stems in point clouds. The feature-based methods utilize spatial and local geometric features of point clouds to identify stems [15,16]. In [17], trunk seeds are identified based on point distribution patterns in each slice. For example, the sizes of canopy slices are always much larger than stem slices. Tao et al. [18] use DBSCAN (Density-Based Spatial Clustering

and Application with Noise) [19] to find stem clusters in sliced point clouds at DBH. Heinzl et al. [20] extract stem points from input TLS data using morphological operations that remove sparse structures like leaves and low vegetation. Stem points can also be recognized based on geometric features such as eigenvalue-based measurements. In [21], the eigenvalue-based geometric features are combined to classify photosynthetic and non-photosynthetic components; then, a group of predefined filters is introduced to remove noise and classification errors. Finally, stem points are grouped into instances based on the distance between non-photosynthetic points. As local features highly depend on the neighborhood sizes, Chen et al. [22] present an adaptive radius selection method based on the scanner-to-point distance, and the detected stem points are merged into tree objects by clustering methods.

A group of top-based tree localization methods is widely used in the processing of airborne and mobile LiDAR point clouds. In general, the top-based method assumes that objects in the real world are always higher than their surroundings, thus the potential objects can be localized by finding local height maxima in point clouds [23]. Li et al. [3] present a new tree segmentation method for airborne LiDAR point clouds where the highest points are treated as treetops. Lin et al. [24] assume that tree crowns can be modeled by circular cones, and propose an individual tree localization method based on multi-scale local maximums in mobile LiDAR point clouds. In [25], treetops in mobile LiDAR data are found by the maximums in the histograms of point numbers. Wang et al. [26] propose a voxel-based tree localization method for mobile LiDAR point clouds, which divides point clouds into 3D voxels, and voxels in the highest layer will be selected as potential treetops. If the distance between potential treetops is smaller than a given threshold, adjacent tree candidates will be grouped into one instance.

In summary, there exist several limitations in the aforementioned tree localization methods for LiDAR point clouds. First, the model-fitting methods may fail if stems are irregular or bifurcated, or point clouds are noisy and suffering from large data missing. Second, many existing methods such as slice-based ones always need the ground as the reference, while the ground modeling is still a challenging task in the complex forest scene. Third, methods that rely on geometric features often involve lots of specific rules or supervised learning processes that may need to be tuned from plots to plots. Last but not least, the top-based methods that are widely used in airborne and mobile LiDAR data processing can not be applied to TLS forest point clouds directly [27] because of the serious data missing in TLS forest point clouds and the complex canopy structures captured by the close scanning. Moreover, the previous top-based object detection methods can not handle overlapped objects.

In this letter, our main contribution is adapting the top-based tree localization method for TLS point clouds of forest scenes. The remainder of this paper is organized as follows: Section 2 elaborates the details of our proposed tree localization method. Section 3 presents the experimental results of the original top-based method and our new method on a TLS forest benchmark [28]. In Section 4, the new method together with compared ones are analyzed. Section 5 summarizes this work.

2. Methods

In the image processing domain, the image enhancement refers to a group of nonlinear refinement techniques which aims at making the images much easier to be interpreted by humans or processed by computers [29,30]. Analogously, to make the tree localization much easier, a new point cloud enhancement method named point cloud inversion (PCI) is proposed. The basic idea of PCI is applying a nonlinear transform to the original point clouds, aiming at enhancing the salience of tree stands in point clouds. The proposed nonlinear transform is inspired by the point cloud inversion operation [31] where point clouds are inverted before approximating the ground surface via the cloth simulation [32]. The detailed steps of the point cloud enhancement method is illustrated as follows.

The original point clouds are first organized into 3D voxels, each of which maintains one index list of points that fall into this voxel. Supposing a voxel can be indexed by its horizontal row, horizontal column and layer (vertical column) number (i, j, k) in the data management and there are m layers for the input scene, an empty index $EI_{i,j}$ for each vertical column indexed by (i, j) is calculated as

$$EI_{i,j} = \sum_{k=0}^{k=m} B(i, j, k), \quad (1)$$

where $B_{i,j,k}$ is a piecewise function which returns 0.0 if the voxel (i, j, k) contains points; otherwise, it returns 1.0. Then, point clouds are transformed according to the following equation:

$$p_{z_{new}}^t \leftarrow \max\{Z_{max} - p_z^t - dV \cdot EI_{i,j}, 0.0\}, \quad (2)$$

where p_t represents a point indexed by t , z is the z coordinate of a point, Z_{max} is the maximal z of the input point clouds, and dV is the voxel resolution. The $Z_{max} - p_z^t$ in Equation (2) stands for the inversion of point clouds, i.e., point clouds will be flipped after the inversion. The $dV \cdot EI_{i,j}$ calculates the length of empty voxels in one vertical column. The updated z value will be large if the original point is close to the ground and its vertical column contains few empty voxels. Otherwise, the updated z value on the left in Equation (2) will be small. The objective of this transformation is enhancing the elevation of vertically connected objects (e.g., stems) while suppressing the elevation of other objects including branches, foliage, low vegetation, and the ground. The idea behinds Equation (2) can be further explained by the demo shown in Figure 1.

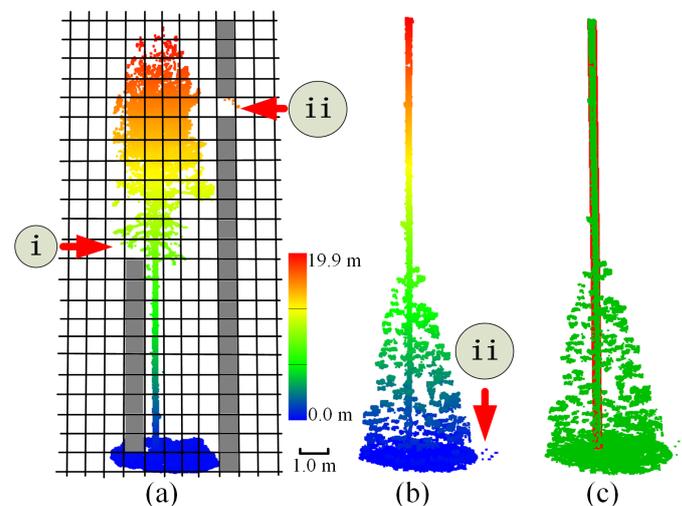


Figure 1. A demo of the PCI transformation. The front view of point clouds are colored by elevation. This toy-example uses 2D grids to represent the side-view of 3D voxels. (a) gray cells will be counted in Equation (1). The arrow i refers to branch points and the arrow ii refers to noisy points; (b) transformed point clouds. (c) The localized tree is indicated by the red vertical line.

In this example, point clouds of one tree instance together with ground are colored by elevation. After dividing the input scene into voxels, i.e., grids in this demo, EI of each vertical column can be calculated. The EI values around tree tops will be small as $Z_{max} - p_z^t$ is close to zero. For root points, their EI values will be large as $Z_{max} - p_z^t$ is large and $dV \cdot EI_{i,j}$ is small. Compared with stem points, the updated z values of branches (e.g., indicated by the arrow i) will be much smaller as there always be many empty voxels in their vertical columns. For noisy points, the updated z values are often small such as points highlighted by the arrow ii. As shown in Figure 1b, it is clear that non-stem points (e.g., branches, noise, and ground) are clustered at a much lower elevation than stem

points which are visually enhanced after the transformation. Finally, the top-based tree localization strategy can be directly applied to localize trees in the enhanced point clouds. The flowchart of top-based tree localization method with the proposed PCI is shown in Figure 2.

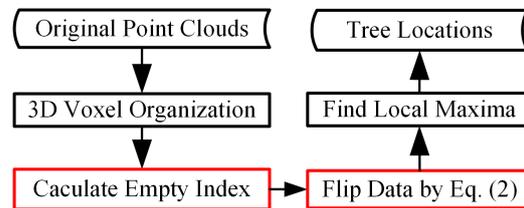


Figure 2. The flowchart of the top-based tree localization method with PCI. Our contribution is highlighted in red.

3. Results

The proposed PCI is tested on the international benchmark for TLS-based forestry applications [28]. The TLS point clouds were collected in 2014 in the southern boreal forest, Finland. In this dataset, the main tree species are Scots pine, Norway spruce and silver, and downy birches. There are six released plots that are 32-by-32 m in size. They are classified into three groups based on the scene complexity (easy, medium, and difficult), and each group consists of two plots. Specifically, plots 1 and 2 are easy plots, plots 3 and 4 belong to medium plots, and plots 5 and 6 are difficult plots. In each plot, there are two kinds of datasets: one single-scan (SS) dataset and one multi-scan (MS) dataset merged from five single scans.

The newly proposed PCI and the original top-based tree localization methods are implemented in C++. The source codes will be released at <https://github.com/GeoVectorMatrix/PointCloudInversion>. The experiments are all conducted on a machine with one i7-6700HQ CPU and 16 GB RAM. The input for our method is original point clouds, and no pre-processing steps such as ground or noise removal are needed. The voxel size is set to 0.25 m based on the experimental tests and a 3×3 window is applied to retrieve local maximums from point clouds in the top-based tree localization step. The experiments are fully automatic with parameters fixed in these plots. The original and transformed point clouds of three represented plots from easy, medium, and difficult categories are shown in Figure 3. The localized trees are visualized by black vertical lines as shown in the enlarged view in Figure 3a. Figure 3b is dominated by red as trees in this plot are about the same height, and their canopies are closely adjacent. After applying PCI, the dominating color turns blue as canopy points are transformed to low points and tree-tops become spatially isolated. In Figure 3c, trees are at different heights, and it is challenging to distinguish canopy instances via visual inspection. With the help of PCI, the distance between tree instances is also increased as shown on the right in Figure 3c. Generally, by visually comparing the point clouds before and after the PCI process, it is clear that point clouds of stems are enhanced while other points are suppressed, enlarging the gaps between adjacent tree instances. Therefore, the top-based methods become much more applicable in localizing tree instances in point clouds. As shown in Figure 3, most tree instances can be localized by the top-based strategy which simply treats the detected local maximums as instance tops.

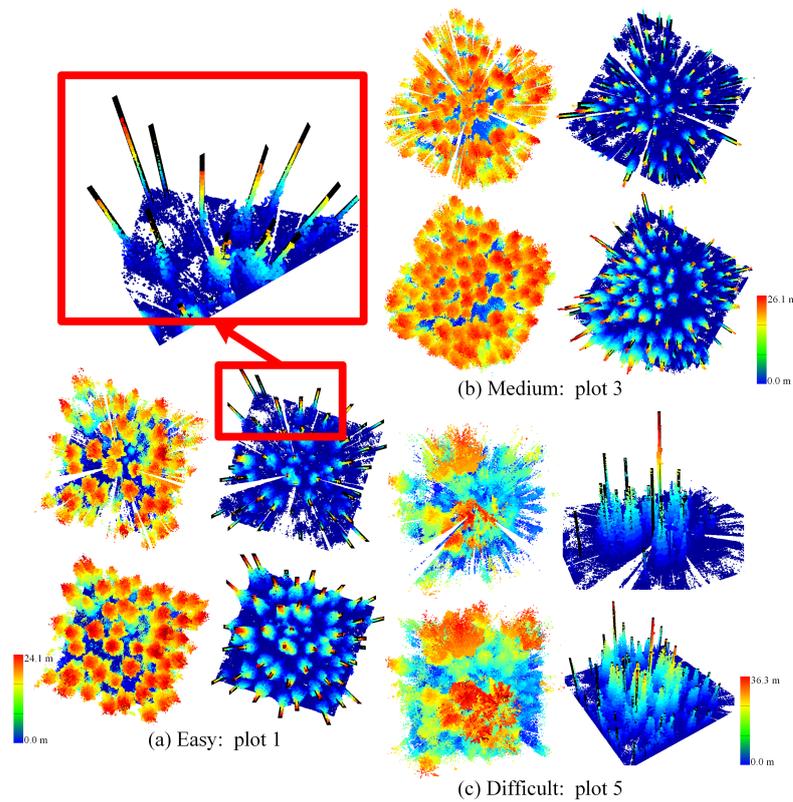


Figure 3. Original point clouds, augmented point clouds, and tree localization results in easy, medium, and difficult plots. In each sub-figure, the first row shows the results of SS and the second row shows the results of MS in that plot. Sub-figures on the left are the original point clouds and the transformed point clouds are on the right. The localized trees are represented by black vertical lines as shown in the enlarged view. Each plot is 32 m-by-32 m in size. (a) results of an easy plot; (b) results of a medium plot; (c) results of a difficult plot.

The completeness and correctness used in [28] are introduced for quantitative evaluations and comparisons:

$$\text{Completeness} = \frac{n_{\text{match}}}{n_{\text{ref}}}, \quad (3)$$

$$\text{Correctness} = \frac{n_{\text{match}}}{n_{\text{extr}}}, \quad (4)$$

where n_{match} is the number of found trees, n_{ref} is the number of reference trees, and n_{extr} is the number of extracted trees. The coordinates of reference trees in all plots are also provided by [28]. The tree localization results with- and without PCI are shown in Figure 4. The objective of this comparative experiment is to demonstrate the necessity and effectiveness of the proposed PCI. The improvement by introducing PCI before the top-based tree localization is significant. In SS tests (Figure 4a,c), both completeness and correctness are almost doubled in nearly all the plots. The results of MS (Figure 4b,d) also show a considerable increase in both completeness and correctness. For instance, the completeness and correctness in easy plots are both lower than 0.4, which rise dramatically to 0.71 and 0.88 after applying PCI, respectively. In the MS datasets of complex scenes, the completeness is nearly trebled from 0.09 to 0.27 as shown in Figure 4b. Overall, this ablation study demonstrates that the newly proposed PCI is helpful in improving the salience of tree instances, making the top-based tree localization framework available in complex forest scenes.

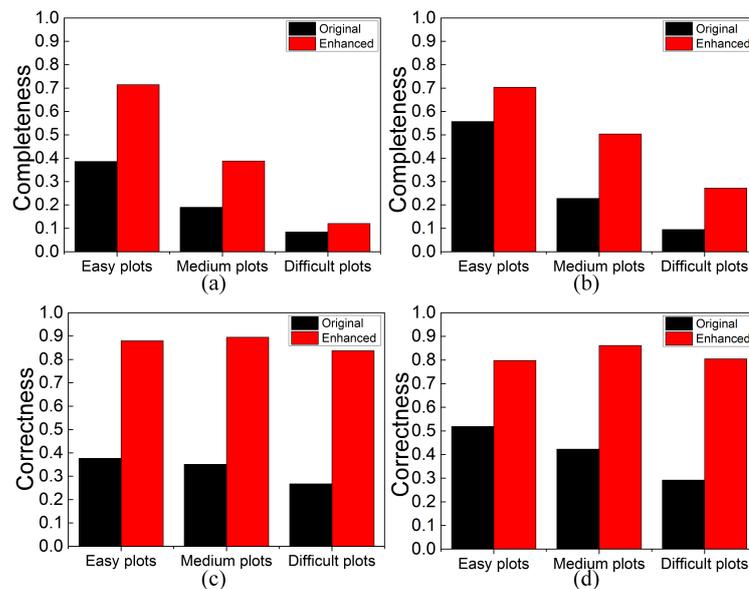


Figure 4. The completeness and correctness of top-based tree localization results with (Enhanced) and without PCI (Original). (a) completeness of tree localization results in SS plots; (b) completeness of tree localization results in MS plots; (c) correctness of tree localization results in SS plots; (d) correctness of tree localization results in MS plots.

The overall performance of the top-based tree localization method with PCI is also calculated and compared with the existing methods. Following the definition in [28], the balanced accuracy measure is defined as

$$Accuracy = \frac{2 * n_{match}}{n_{ref} + n_{extr}}. \quad (5)$$

The mean accuracies of our results in all plots without- and with PCI are shown in Figure 5. A quick conclusion is that there is a dramatic increase in the accuracy by adapting PCI for the top-based method. Results of four baseline methods named TUDeft, InraIGN, TreeMetrics, and NJU, which are reported in [28] are also drawn in Figure 5. The goal of this comparative experiment is to demonstrate the ability and potential of the top-based methods in localizing trees from TLS point clouds.

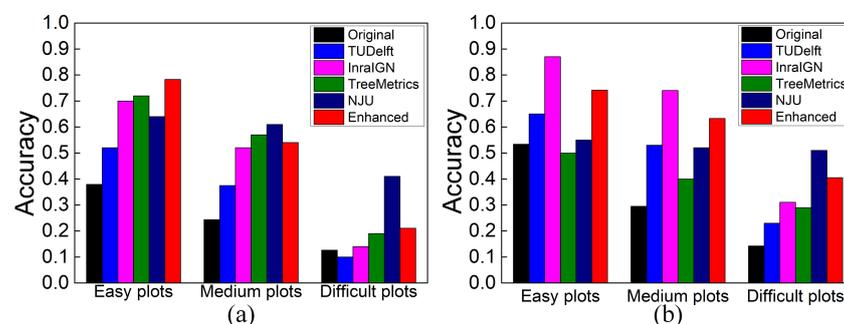


Figure 5. The accuracy of tree localization results of our methods and other baseline methods. (a) the accuracy in SS datasets; (b) the accuracy in MS datasets.

The TUDeft uses a voxel-wise clustering method which merges point clouds into different instances. The InraIGN method builds ground reference based on the identified ground points; then, a horizontal slice of point clouds at DBH is extracted to find tree candidates. The TreeMetrics method also removes ground points first and tree candidates

are detected by clustering of sliced points and model fitting. The NJU presents a supervised learning method that classifies original point clouds into scattering, linear, and surface types. After applying filters to remove branches and noise, stems can be identified from linear points. In most SS and MS datasets, the accuracy of the top-based method with PCI ranks the second among all six methods while the ranked-first method changes with the plot complexity and scanning modes. Specifically, the top-based method with PCI ranks first in the easy plot in SS datasets. As the original top-based strategy is not applicable in TLS forest point clouds, this improved performance is attributed to the proposed PCI.

4. Discussion

The experiments and results in Section 3 demonstrate the necessity, effectiveness, and stability of the proposed PCI. Although the top-based method relying on PCI can achieve comparable results with several baseline methods, it should be noticed that its performance is not as good as the top-ranked methods reported in [28]. However, the goal of our research is not achieving the highest accuracy in some specific benchmarks. The initial objective is introducing a new preprocessing framework which facilitates the data processing steps such as pole-like object detection. By introducing PCI, the top-based method which is rarely used in TLS forest data processing becomes applicable and can achieve competitive results.

A real example is given in Figure 6 to show the advantages of the PCI-aided top-based localization method for TLS forest scenes. There are two tree instances in Figure 6, the left one is tall and the right one is under its canopy. By using the original top-based method, multiple local tops can be identified due to their complex structures and noise. In addition, it is impossible to localize overlapped objects by the original top-based methods. As shown in Figure 6a, two vertical lines representing two tree top candidates are drawn, and the right dashed line indicates a wrong candidate. By applying PCI, the top-based method can find two correct tree instances as shown in Figure 6c. Another advantage of the PCI-aided top-based method lies in its low dependence on priors such as the accurate ground surface and geometric shapes of the stem. For example, the model fitting method can only find one correct circle in the upper-left corner in the sliced point clouds as shown in Figure 6b, making it impossible to localize the small tree. Moreover, both the PCI and the top-based method are fast and easy to be implemented, while the time-consuming data processing steps such as ground modeling and feature calculation are not involved.

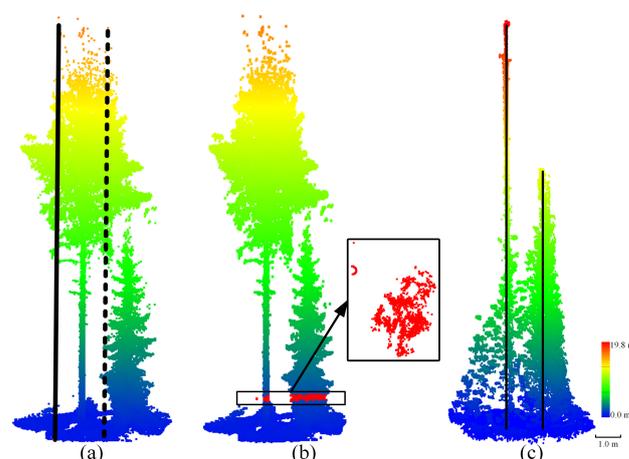


Figure 6. Comparisons between existing tree localization methods and our PCI-based method. (a) multiple local maximums within one tree instance; (b) sliced point clouds at DBH; (c) the results of our method.

The usage of the proposed PCI is not limited to the tree localization in forest. We apply PCI to TLS point clouds of a campus scene to find pole-like objects. In the upper row of Figure 7a, the open-access point clouds provided by [33] are colored by elevation,

and the pole light among roadside trees is circled by a black rectangle. In the second row of Figure 7a, pole-like objects are localized by the top-based method aided by PCI. This test shows that the PCI is flexible and potential for localizing other objects in various point clouds. As shown in Figure 7a, many trees are surrounded by tree supports, increasing the difficulty of using model fitting or feature-based methods. In addition, due to the data incompleteness at DBH, the slice-based method may fail. As shown in Figure 7b, tree instances that are surrounded by tree supports or suffer from data missing at DBH are all visually enhanced by the PCI. Thus, the proposed method is robust against complex structures and missing data. Furthermore, the top-based method used in this letter is a simple implementation that can be improved by adaptive window sizes or combining other advanced algorithms [3,24,34]. Therefore, there is much room for improvement in the PCI-aided top-based methods.

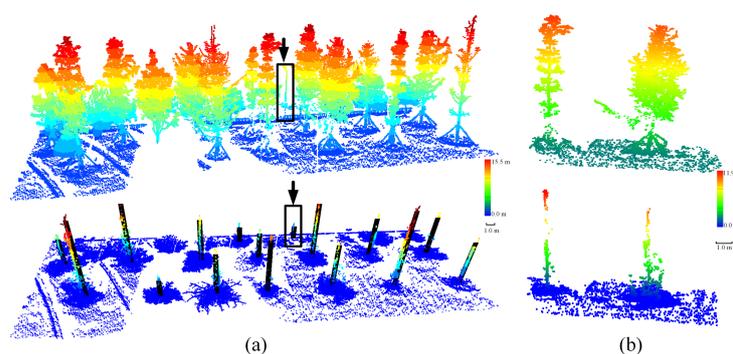


Figure 7. The potential of localizing street trees by the PCI-based method. (a) tree localization in TLS point clouds of street trees in a campus. The pole-like object in the rectangle is also identified by the PCI-based method. (b) The PCI can enhance trees with different structures and is robust against the data incompleteness.

5. Conclusions

In this letter, we propose a new point cloud enhancement method named PCI that makes the existing top-based object localization framework applicable in TLS forest scenes for the first time. The PCI is easy to implement and requires no ground reference or noise removal. Furthermore, PCI eliminates the need of the time-consuming feature calculation in other methods. In addition, the PCI can localize overlapped tree instances and other man-made pole-like objects in different scenes like the campus. The authors believe that the PCI also provides a new perspective for point cloud processing. For example, the existing object detection methods such as slice-based and model-fitting based ones may also benefit from the PCI. Our future work will focus on exploring potential applications based on the nonlinear transformations of point clouds.

Author Contributions: Conceptualization, S.X. (Shaobo Xia); Methodology, S.X. (Shaobo Xia) and S.X. (Sheng Xu); Writing—original draft preparation, S.X. (Shaobo Xia), D.C., S.X. (Sheng Xu), P.W., and J.P.; Writing—review and editing, J.P. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Natural Science Foundation of Jiangsu Province, Grant No. BK20200784, and the National Natural Science Foundation of China, Grant No. 41971415, and the China Postdoctoral Science Foundation, Grant No. 2019M661852, and the Talent Startup Project of Zhejiang A & F University Scientific Research Development Foundation, Grant No. 2034020104.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The benchmark datasets used in this paper can be found from: <http://laserscanning.fi/tls-benchmarking-results/> and <http://3s.whu.edu.cn/ybs/en/benchmark.htm>.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

| | |
|--------|---|
| 3D | Three-Dimensional |
| ALS | Airborne Laser Scanning |
| DBH | Diameter at Breast Height |
| DBSCAN | Density-Based Spatial Clustering and Application with Noise |
| PCI | Point Cloud Inversion |
| SS | Single Scans |
| TLS | Terrestrial Laser Scanning |
| MS | Multi-Scans |

References

1. Wang, D. Unsupervised semantic and instance segmentation of forest point clouds. *ISPRS J. Photogramm. Remote Sens.* **2020**, *165*, 86–97. [[CrossRef](#)]
2. Zhang, W.; Wan, P.; Wang, T.; Cai, S.; Chen, Y.; Jin, X.; Yan, G. A novel approach for the detection of standing tree stems from plot-level terrestrial laser scanning data. *Remote Sens.* **2019**, *11*, 211. [[CrossRef](#)]
3. Li, W.; Guo, Q.; Jakubowski, M.K.; Kelly, M. A new method for segmenting individual trees from the lidar point cloud. *Photogramm. Eng. Remote Sens.* **2012**, *78*, 75–84. [[CrossRef](#)]
4. Lu, X.; Guo, Q.; Li, W.; Flanagan, J. A bottom-up approach to segment individual deciduous trees using leaf-off lidar point cloud data. *ISPRS J. Photogramm. Remote Sens.* **2014**, *94*, 1–12. [[CrossRef](#)]
5. Saarinen, N.; Kankare, V.; Vastaranta, M.; Luoma, V.; Pyörälä, J.; Tanhuanpää, T.; Liang, X.; Kaartinen, H.; Kukko, A.; Jaakkola, A.; et al. Feasibility of Terrestrial laser scanning for collecting stem volume information from single trees. *ISPRS J. Photogramm. Remote Sens.* **2017**, *123*, 140–158. [[CrossRef](#)]
6. Liang, X.; Litkey, P.; Hyyppä, J.; Kaartinen, H.; Vastaranta, M.; Holopainen, M. Automatic stem mapping using single-scan terrestrial laser scanning. *IEEE Trans. Geosci. Remote Sens.* **2012**, *50*, 661–670. [[CrossRef](#)]
7. Rutzinger, M.; Pratihast, A.K.; Oude Elberink, S.J.; Vosselman, G. Tree modelling from mobile laser scanning data-sets. *Photogramm. Rec.* **2011**, *26*, 361–372. [[CrossRef](#)]
8. Wang, Z.; Zhang, L.; Fang, T.; Mathiopoulos, P.T.; Qu, H.; Chen, D.; Wang, Y. A structure-aware global optimization method for reconstructing 3D tree models from terrestrial laser scanning data. *IEEE Trans. Geosci. Remote Sens.* **2014**, *52*, 5653–5669. [[CrossRef](#)]
9. Dai, W.; Yang, B.; Liang, X.; Dong, Z.; Huang, R.; Wang, Y.; Pyörälä, J.; Kukko, A. Fast registration of forest terrestrial laser scans using key points detected from crowns and stems. *Int. J. Digit. Earth* **2020**, *13*, 1585–1603. [[CrossRef](#)]
10. Zheng, G.; Moskal, L.M. Computational-geometry-based retrieval of effective leaf area index using terrestrial laser scanning. *IEEE Trans. Geosci. Remote Sens.* **2012**, *50*, 3958–3969. [[CrossRef](#)]
11. Seidel, D.; Albert, K.; Ammer, C.; Fehrmann, L.; Kleinn, C. Using terrestrial laser scanning to support biomass estimation in densely stocked young tree plantations. *Int. J. Remote Sens.* **2013**, *34*, 8699–8709. [[CrossRef](#)]
12. Ye, W.; Qian, C.; Tang, J.; Liu, H.; Fan, X.; Liang, X.; Zhang, H. Improved 3D Stem Mapping Method and Elliptic Hypothesis-Based DBH Estimation from Terrestrial Laser Scanning Data. *Remote Sens.* **2020**, *12*, 352. [[CrossRef](#)]
13. Shao, J.; Zhang, W.; Mellado, N.; Wang, N.; Jin, S.; Cai, S.; Luo, L.; Lejemble, T.; Yan, G. SLAM-aided forest plot mapping combining terrestrial and mobile laser scanning. *ISPRS J. Photogramm. Remote Sens.* **2020**, *163*, 214–230. [[CrossRef](#)]
14. Maas, H.G.; Bienert, A.; Scheller, S.; Keane, E. Automatic forest inventory parameter determination from terrestrial laser scanner data. *Int. J. Remote Sens.* **2008**, *29*, 1579–1593. [[CrossRef](#)]
15. Ritter, T.; Schwarz, M.; Tockner, A.; Leisch, F.; Nothdurft, A. Automatic mapping of forest stands based on three-dimensional point clouds derived from terrestrial laser-scanning. *Forests* **2017**, *8*, 265. [[CrossRef](#)]
16. Xia, S.; Wang, C.; Pan, F.; Xi, X.; Zeng, H.; Liu, H. Detecting stems in dense and homogeneous forest using single-scan TLS. *Forests* **2015**, *6*, 3923–3945. [[CrossRef](#)]
17. Li, L.; Li, D.; Zhu, H.; Li, Y. A dual growing method for the automatic extraction of individual trees from mobile laser scanning data. *ISPRS J. Photogramm. Remote Sens.* **2016**, *120*, 37–52. [[CrossRef](#)]
18. Tao, S.; Wu, F.; Guo, Q.; Wang, Y.; Li, W.; Xue, B.; Hu, X.; Li, P.; Tian, D.; Li, C.; et al. Segmenting tree crowns from terrestrial and mobile LiDAR data by exploring ecological theories. *ISPRS J. Photogramm. Remote Sens.* **2015**, *110*, 66–76. [[CrossRef](#)]
19. Ester, M.; Kriegel, H.P.; Sander, J.; Xu, X. A density-based algorithm for discovering clusters in large spatial databases with noise. In *Kdd; AAAI: Portland, OR, USA, 1996; Volume 96*, pp. 226–231.
20. Heinzel, J.; Huber, M.O. Detecting tree stems from volumetric TLS data in forest environments with rich understory. *Remote Sens.* **2016**, *9*, 9. [[CrossRef](#)]

21. Ma, L.; Zheng, G.; Eitel, J.U.; Moskal, L.M.; He, W.; Huang, H. Improved salient feature-based approach for automatically separating photosynthetic and nonphotosynthetic components within terrestrial lidar point cloud data of forest canopies. *IEEE Trans. Geosci. Remote Sens.* **2016**, *54*, 679–696. [[CrossRef](#)]
22. Chen, M.; Wan, Y.; Wang, M.; Xu, J. Automatic stem detection in terrestrial laser scanning data with distance-adaptive search radius. *IEEE Trans. Geosci. Remote Sens.* **2018**, *56*, 2968–2979. [[CrossRef](#)]
23. Golovinskiy, A.; Kim, V.G.; Funkhouser, T. Shape-based recognition of 3D point clouds in urban environments. In Proceedings of the 2009 IEEE 12th International Conference on Computer Vision, Kyoto, Japan, 29 September–2 October 2009; IEEE: Piscataway, NJ, USA, 2009; pp. 2154–2161.
24. Lin, Y.; Hyypä, J.; Jaakkola, A.; Yu, X. Three-level frame and RD-schematic algorithm for automatic detection of individual trees from MLS point clouds. *Int. J. Remote Sens.* **2012**, *33*, 1701–1716. [[CrossRef](#)]
25. Zhong, L.; Cheng, L.; Xu, H.; Wu, Y.; Chen, Y.; Li, M. Segmentation of individual trees from TLS and MLS Data. *IEEE J. Sel. Top. Appl. Earth Observ. Int Remote Sens* **2016**, *99*, 1–14. [[CrossRef](#)]
26. Wang, J.; Lindenbergh, R.; Menenti, M. Scalable individual tree delineation in 3D point clouds. *Photogramm. Rec.* **2018**, *33*, 315–340. [[CrossRef](#)]
27. Xi, Z.; Hopkinson, C.; Chasmer, L. Automating plot-level stem analysis from terrestrial laser scanning. *Forests* **2016**, *7*, 252. [[CrossRef](#)]
28. Liang, X.; Hyypä, J.; Kaartinen, H.; Lehtomäki, M.; Pyörälä, J.; Pfeifer, N.; Holopainen, M.; Broly, G.; Francesco, P.; Hackenberg, J.; et al. International benchmarking of terrestrial laser scanning approaches for forest inventories. *ISPRS J. Photogramm. Remote Sens.* **2018**, *144*, 137–179. [[CrossRef](#)]
29. Beutel, J.; Kundel, H.L.; Van Metter, R.L. *Handbook of Medical Imaging*; SPIE Press: Bellingham, WA USA, 2000; Volume 1.
30. Arce, G.R.; Bacca, J.; Paredes, J.L. Nonlinear filtering for image analysis and enhancement. In *The Essential Guide to Image Processing*; Elsevier: Amsterdam, The Netherlands, 2009; pp. 263–291.
31. Zhang, W.; Qi, J.; Wan, P.; Wang, H.; Xie, D.; Wang, X.; Yan, G. An easy-to-use airborne LiDAR data filtering method based on cloth simulation. *Remote Sens.* **2016**, *8*, 501. [[CrossRef](#)]
32. Weil, J. The synthesis of cloth objects. *ACM Siggraph Comput. Graph.* **1986**, *20*, 49–54. [[CrossRef](#)]
33. Dong, Z.; Liang, F.; Yang, B.; Xu, Y.; Zang, Y.; Li, J.; Wang, Y.; Dai, W.; Fan, H.; Hyypä, J.; et al. Registration of large-scale terrestrial laser scanner point clouds: A review and benchmark. *ISPRS J. Photogramm. Remote Sens.* **2020**, *163*, 327–342. [[CrossRef](#)]
34. Chen, Q.; Baldocchi, D.; Gong, P.; Kelly, M. Isolating individual trees in a savanna woodland using small footprint lidar data. *Photogramm. Eng. Remote Sens.* **2006**, *72*, 923–932. [[CrossRef](#)]