



Individual stem detection in residential environments with MLS data

Sheng Xu, Shanshan Xu, Ning Ye & Fa Zhu

To cite this article: Sheng Xu, Shanshan Xu, Ning Ye & Fa Zhu (2018) Individual stem detection in residential environments with MLS data, Remote Sensing Letters, 9:1, 51-60, DOI: [10.1080/2150704X.2017.1384588](https://doi.org/10.1080/2150704X.2017.1384588)

To link to this article: <https://doi.org/10.1080/2150704X.2017.1384588>



Published online: 26 Oct 2017.



Submit your article to this journal [↗](#)



Article views: 167



View Crossmark data [↗](#)



Citing articles: 2 View citing articles [↗](#)



Individual stem detection in residential environments with MLS data

Sheng Xu^a, Shanshan Xu^a, Ning Ye^a and Fa Zhu^b

^aCollege of Information Science and Technology, Nanjing Forestry University, Nanjing, China; ^bSchool of Computer Science and Engineering, Nanjing University of Science and Technology, Nanjing, China

ABSTRACT

Nowadays, mobile laser scanning (MLS) system succeeds to collect plentiful side information of roadside trees. This letter aims to propose a method for the individual stem detection in residential environments using mobile LiDAR point clouds. The first step is to use the proposed point removal method for filtering ground points based on the elevation histogram. The second step is to localize trees by using a circle fitting method on the projection points of trunks. The third step is to use the voxel-based representation approach to organize points and calculate the voxel value for the following optimization. The last step is to minimize the formulated stem energy function by the dynamic programming technique. The main contribution of our work is that the cost of finding a stem is calculated by a penalty function and minimized by a dynamic programming model. Test on residential environments shows that our method achieves the completeness of 94.2% and correctness of 95.7%, which are competitive results in the stem detection.

ARTICLE HISTORY

Received 5 June 2017

Accepted 18 September 2017

1. Introduction

Individual tree detection from point clouds finds various applications in forestry, such as the canopy segmentation (Lu et al. 2014), species classification (Matsuki, Yokoya, and Iwasaki 2015) and vegetation analysis (Yun et al. 2016). Airborne laser scanning has been used in the information collection for the forest research several years ago, however, airborne LiDAR point clouds are usually in a low density and difficult to contain the side information of roadside trees. Recently, mobile laser scanning (MLS) data, which are collected with a high point density in a relatively lower cost, are becoming available. MLS succeeds to provide abundant tree side information and brings more chances for the forestry. In the following, several typical methods proposed for the stem detection with LiDAR data will be reviewed.

Hetti Arachchige (2013) used the principal direction and geometric shapes as geometric descriptors in the stem detection. They did not need any prior knowledge before the detection and worked well in various kinds of tree structures. However, their stem growing process was easy to be affected by data gaps along the stem. Lehtomäki et al. (2010) extracted trunks from mobile LiDAR data by proposing a detection framework, including the

scan line segmentation, groups clustering, clusters merging and candidate clusters classification. However, when a trunk was inside branches, they were difficult to extract the stem. Liang et al. (2014) proposed a model-based method to estimate the location of the stem by using a series of 3D cylinders. They achieved a high performance in the test forest where most stems were oriented vertically. Xia et al. (2015) detected stems by proposing a classifier using multi-scale geometric features. In their work, there was no requirement of the circle or cylinder fitting process, however, their results highly depended on the accuracy of the scene classification. Guan et al. (2015) proposed a voxel-based segmentation method to detect trees from mobile LiDAR point clouds. They succeeded to segment overlapping trees into individual objects. In the method of Zhong et al. (2017), they provided an approach based on the histogram of octree nodes, and used the kurtosis threshold and ground connectivity as constraints to distinguish tree stems in mobile LiDAR point clouds. However, neither Guan et al. (2015) nor Zhong et al. (2017) showed results of the stem detection in their work. The motivation of this letter is to obtain optimal stems from mobile LiDAR point clouds and provide a high accuracy of the stem detection in residential environments.

1.1. Ground point removal

Although existing point filtering methods are proved to work well in airborne LiDAR point clouds, they may achieve a poor performance in our MLS data. For example, Progressive triangular irregular network densification (Axelsson 2000), which is a state-of-the-art ground point removal method, appears problems. Points which are close to the ground can be misclassified as ground points. This is because that a user-defined threshold for the judgment of non-ground points is difficult to be adaptive for different environments.

In this section, we propose a new point removal method for filtering ground points adaptively based on the elevation information. Since ground points are much denser than the off-ground points in MLS data, there exists a peak in the elevation histogram to indicate the region of ground points. The expected histogram of mobile LiDAR point clouds is shown in Figure 1, including a peak point (x_p, y_p) and a terminal point (x_t, y_t) . The elevation threshold for the judgment of the off-ground and ground is supposed to be between the peak point and the terminal point in the histogram. In our method, we use a moving point (x_m, y_m) for searching the optimal elevation in ground filtering. Denote the angle between the vector $\mathbf{MP} = (x_p - x_m, y_p - y_m)$ and $\mathbf{MT} = (x_t - x_m, y_t - y_m)$ as φ ,

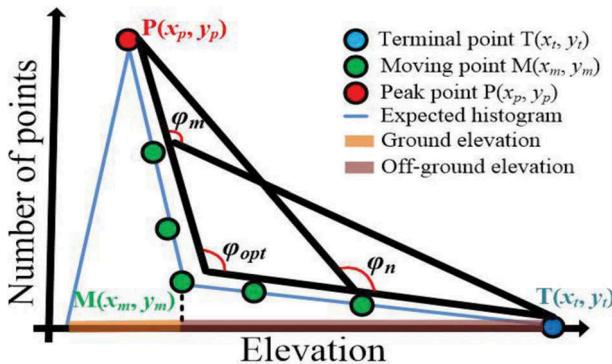


Figure 1. Illustration of the ground point removal.

which is used to illustrate the change rate of points' elevation. The moving point, which achieves the smallest φ , will be chosen to indicate the optimal elevation for the ground removal. As shown in Figure 1, either a lower or a higher elevation will cause a large angle φ_m or φ_n compared with the angle φ_{opt} .

Figure 2 shows the example of our proposed ground removal in MLS data. In this case, the elevation histogram of the input scene is shown in Figure 3(a). There is more than one peak in the histogram. The goal is to find the smallest angle between the vector \mathbf{MP}_i and \mathbf{MT} , where $i = 1$ or 2 , as shown in Figure 3(b). In our removal, the valid smallest angle is required to be greater than $\pi/2$. The optimal threshold is decided by the moving point which is 11.20 meters in this example. Our method does not require any user-defined threshold and can be adaptive to most residential environments. Problems may appear in mountains and data are required to be split into small pieces to keep the elevation of the ground consistently.

1.2. Tree localization

In this section, we will localize trees by using a circle fitting method on the projection of trunks. The selected regions for projecting are points at a height of 120–160 cm above the ground. Below are details of our localization.

First, extract those points that at a height of 120–160 cm above the detected ground as shown in Figure 4(a). Second, project the above extracted points onto a 2D plane as shown in Figure 4(b). Third, use a circle fitting method to localize candidate trees. In the fitting process,

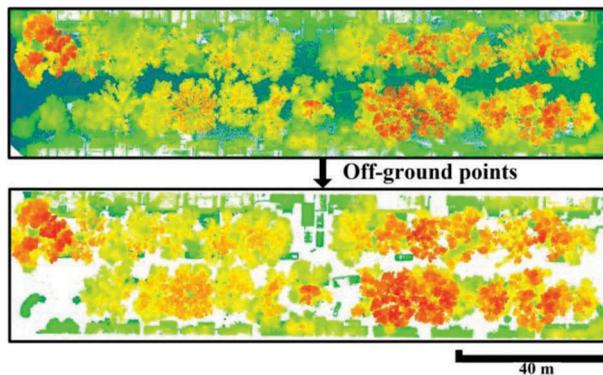


Figure 2. Performance of the removal method.

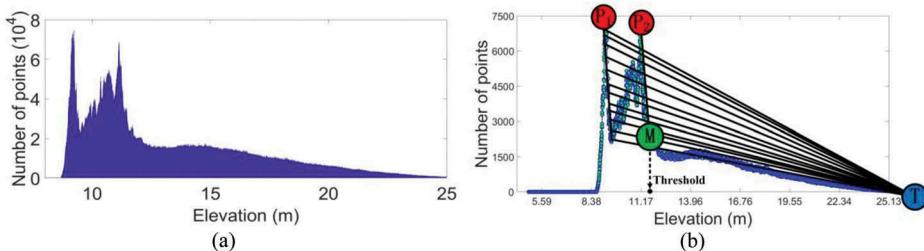


Figure 3. Illustration of obtaining the optimal threshold value. (a) Elevation histogram of the input scene. (b) The search for the optimal threshold value.

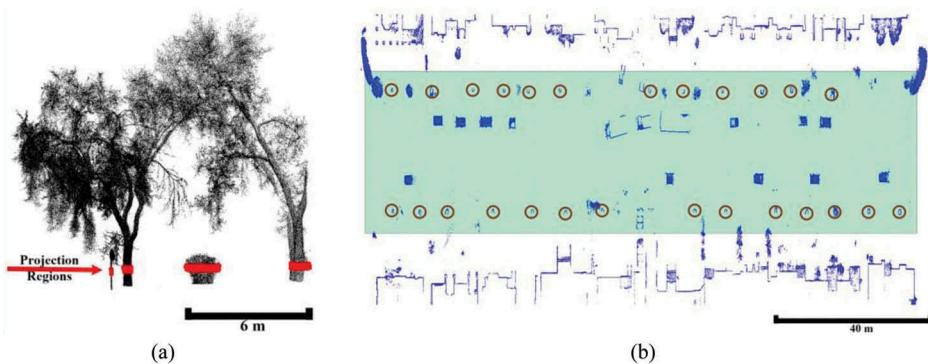


Figure 4. Performance of the tree localization. (a) Projection regions. (b) Localization of candidate trunks.

we choose the random sample consensus (RANSAC) (Schnabel, Wahl, and Klein 2007) to estimate parameters of circles from our point clouds. RANSAC has a high robustness for incomplete data containing outliers and works well in MLS data. Circles in Figure 4(b) show localization results of trees from the projection plane. The centre point of each circle will be chosen as a seed and we will conduct our optimization for all valid seeds to find stems.

1.3. Voxel based organization

Mobile LiDAR point clouds are unorganized, massive and uneven. In this section, we propose a voxel-based representation method to organize points. Input points are split into adjacent $1\text{ cm} \times 1\text{ cm} \times 1\text{ cm}$ cubes which are called voxels. The key is how to assign the voxel value for the optimization of the stem growing.

In our work, the stem is supposed to have a consistent direction in growing. Therefore, the voxel value is formulated to show the growing direction of the current point. In order to obtain the growing direction of points, we define the optimal projection of each point. The optimal plane is calculated by projecting neighbouring points to planes with different normal vectors. The projection process is achieved by using the spherical coordinate system. The normal vector of each plane is calculated by $(\sin\phi \times \cos\theta, \sin\phi \times \sin\theta, \cos\phi)$. θ and ϕ are angles required in the spherical polar coordinates and they range from 0 to 2π and 0 to π , respectively. The projection plane with the largest density will be chosen as the optimal plane of the current point. The points' density is defined by the mean Euclidean distance between the current point and its neighbouring points. In our work, we split the angle θ and ϕ into nine bins. The interval of the angle θ and ϕ is $2\pi/9$ and $\pi/9$, respectively. Thus, there are 81 candidate projection planes. The value of a voxel depends on the ϕ of the projection plane. For a projection plane, if $\phi \in [3\pi/9, 4\pi/9)$, the value of the voxel is 3 as shown in Figure 5. The value of the voxel is calculated as centrosymmetric.

1.4. The stem growing

In the stem growing, we define the ROI (region of interest) as a cylinder as shown in Figure 6. The diameter of the cylinder is 2 meters and the length is the height of the input tree. The stem of a tree will be built by voxels in ROI with the minimum cost from

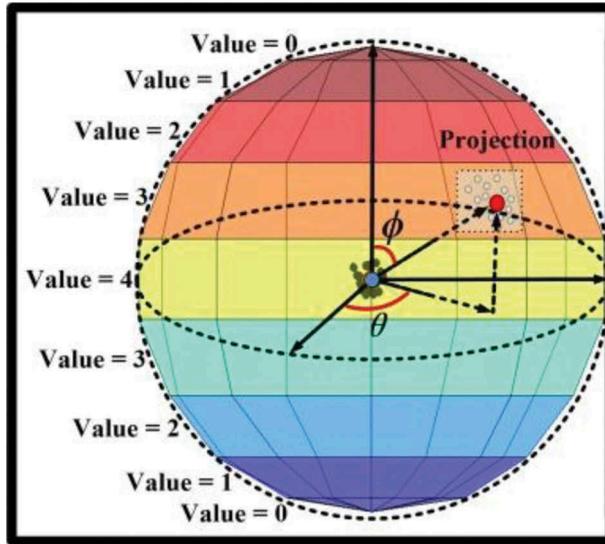


Figure 5. Illustration of the spherical projection method.

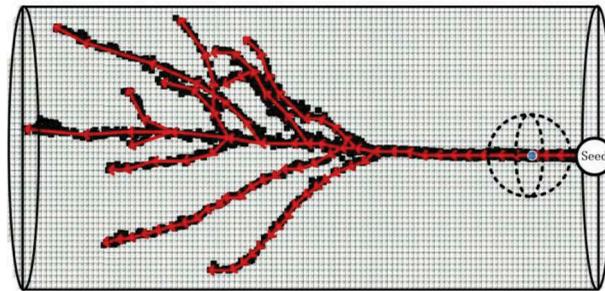


Figure 6. Illustration of the ROI (region of interest) and the stem growing.

the seed point. Following shows the formulation of an energy function for calculating the cost of a stem and the minimization of the proposed object function.

1.4.1. Formulation of the stem cost

Our energy function uses a data term and a smoothness term to represent the cost of different stems and can be defined as:

$$E_N = \sum_i^N (D_i(x, y, z) + S(i, j)) \quad (1)$$

where N is the number of voxels in the stem and (x, y, z) is the coordinate of the current voxel i . Voxel j is the prior voxel of i in the stem. The data term D refers to the cost of a stem containing all selected voxels and the smoothness term S is the cost of connecting the voxel i with j in the stem. The above-mentioned two terms are calculated as

$$D_i(x, y, z) = \begin{cases} 0, & i \in C \\ \beta, & i \notin C \end{cases} \quad (2)$$

$$S(i, j) = \begin{cases} d(i, j), & V(i) = V(j) \\ |V(i) - V(j)| \times d(i, j), & V(i) \neq V(j) \end{cases}$$

In the data term, C is the set of voxels that containing points. If a stem contains an empty voxel, i.e., a voxel has no points inside, there will be a preset penalty β . In the smoothness term, $V(i)$ and $V(j)$ are the value of the voxel i and j , respectively. Function $d(i, j)$ is to calculate the Euclidean distances between voxels i and j . If voxels i and j are in different growing directions, the stem containing both i and j will be assigned a large penalty. Voxels in the ROI formulate candidate stems and our task is to find the one whose cost achieves the minimum energy.

1.4.2. Optimization of the stem growing

To achieve the minimization of E_N , we model a graph using voxels as our nodes. The left of Figure 7 shows the input point clouds and the middle of Figure 7 demonstrates the formulated set C using our voxel-based representation. The right of Figure 7 displays the modeled graph filled with empty voxels and the set C voxels. By adding a source node A and a sink node B , each stem is regarded as a path from A to B . Connections between adjacent voxels are weighted by the sum of the data term and smoothness term. The capacity of a path from A to B is the cost of the stem. Our goal is to find the path of the minimum cost. The number of candidate paths is large. It is impossible to find the optimal path by enumerating the cost of all paths. The following shows how to solve the optimal path by the dynamic programming (DP) method efficiently.

DP is a well-known optimization method for solving the minimum cost. The idea is to break the complex problem down into a collection of simpler sub-problems. When a path adds a new node q , the optimal path from the starting node A to q must contain the solution from the starting node A to q 's prior node p . The sub-problem of our optimization is to find an optimal path from p to the newly added node q . For each node p , we store the optimal path from A to p and the cost E_p of the path. When adding a new node q to the path, the required optimization is the minimization of Equation (3).

$$E_q = \min_{p \in Q} (E_p + D_q(x, y, z) + S(q, p)) \quad (3)$$

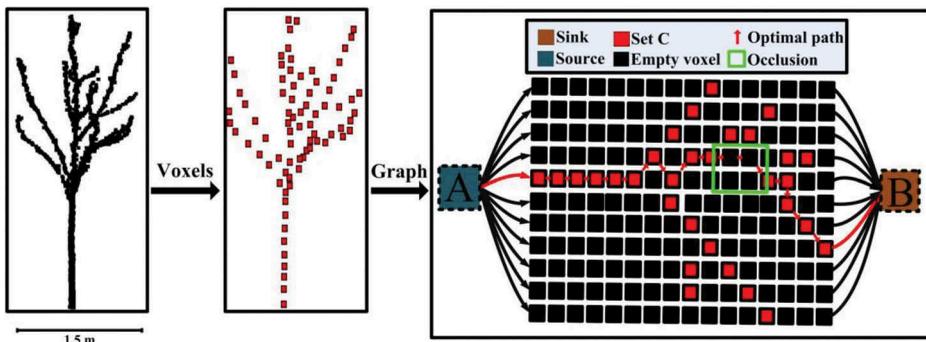


Figure 7. Illustration of the graph to be optimized, including input tree points, voxels in the set C and graph for the optimization.

where Q is the set of adjacent nodes of q . When q arrives at the sink node B , we finish the calculation of all paths' cost in the current ROI. The path of the least cost will be chosen as the optimal stem for this ROI. It is worthy to mention that even the stem is incomplete, our method can figure out the optimal stem by calculating the penalty of containing empty voxels in the path as shown in the right of [Figure 7](#).

2. Experiments and discussion

2.1. Dataset and parameters

The test data is an area containing 190,607,045 points acquired by the RiegI VMX system. The test area is located at a residential area (48°26'05N, 123°20'50W) of Victoria, BC, Canada. All required parameters in our stem detection are shown in [Table 1](#). In the value of θ and ϕ , n is an integer and ranges from 0 to 8.

2.2. Evaluation

In our work, the evaluation of the stem detection is on both the individual stem level and point level. Results of the stem detection based on the individual stem level are shown in [Figure 8\(a\)](#). A close view of our detected stems is shown in [Figure 8\(b\)](#). There are 129 visible stems in the input points and 120 stems are detected by the proposed method which achieves an accuracy of 93.02%.

There are three challenging cases in our stem detection as shown in [Figure 9](#). (1) Pole-like objects (e.g., the streetlight and pole) which are easy to be wrongly detected as trees in the localization. The solution is to analyse the distribution of points in the ROI using the Kurtosis method (Zhong et al. 2017) to remove false trunks before the stem detection. (2) Trees without visible stem points (e.g., the pine tree) or there are artefacts (e.g., the hedge or fence) or understory shrubs around the trees. In order to obtain all seeds for growing trees in this case, we set a large error tolerance in RANSAC. This causes the appearance of over-sampling circles for indicating starting points. Although there are wrongly extracted circles, we can remove them in the analysis of the point distribution in the ROI. (3) Trees with a small diameter (e.g., the thin tree) at the height of 120–160 cm above the ground which are easy to be missed. Our suggestion is to set a specific small diameter (5 cm) threshold in the RANSAC for these trees.

For the evaluation on the point level, results of the detection include the true positive (TP), false negative (FN) and false positive (FP) (Hetti Arachchige 2013). The required

Table 1. Parameters in the proposed stem detection.

Process	Parameters	
	Description	Value
Tree localization	Diameter of the trunk	>20 cm
	Height of the tree	>2 m
	Elevation of points to be projected	120–160 cm
Voxel based organization	Generated voxel size	1 cm × 1 cm × 1 cm
	Diameter of the cylinder	2 m
	The value of θ	$n \times 2\pi/9$ rad
	The value of ϕ	$n \times \pi/9$ rad
Stem growing	Coefficient β in data term	1

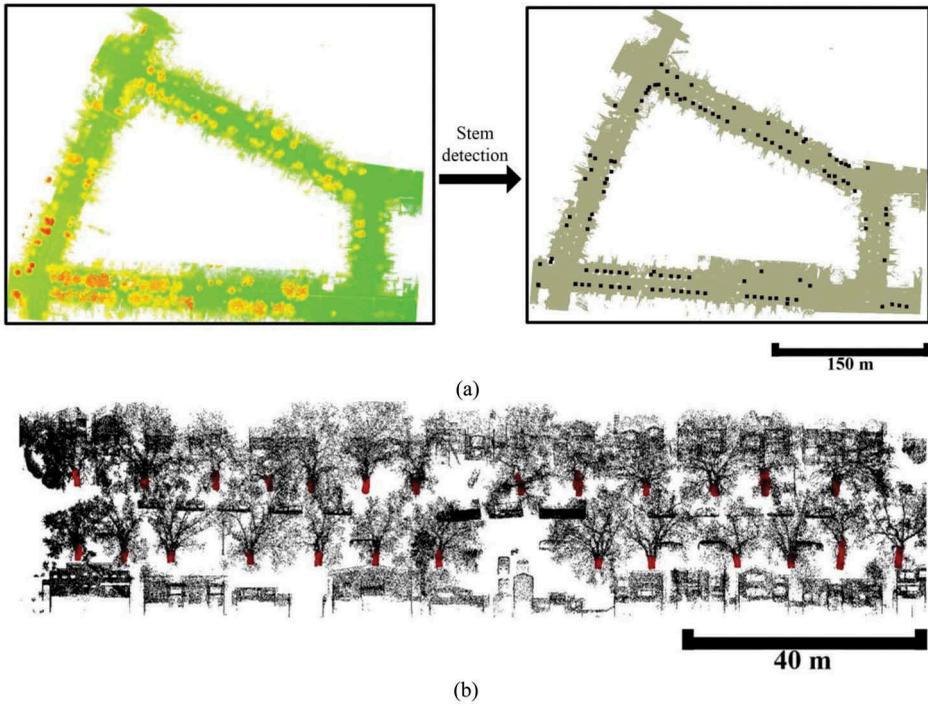


Figure 8. Demonstrate of detected stems. (a) Input point clouds and detected stems. (b) Details of the detected stems.

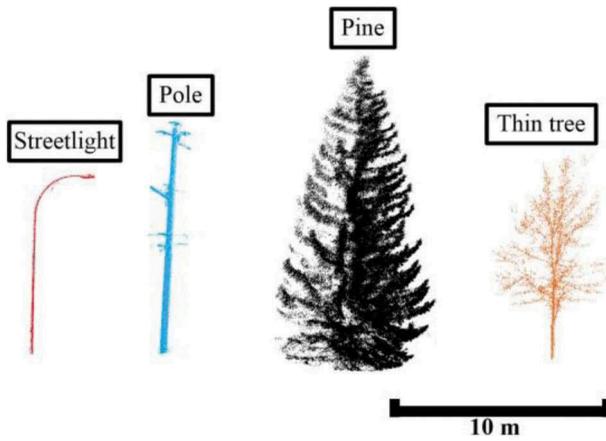


Figure 9. Challenging cases. (a) Streetlight. (b) Pole. (c) Pine. (d) Shrub. (e) Thin tree.

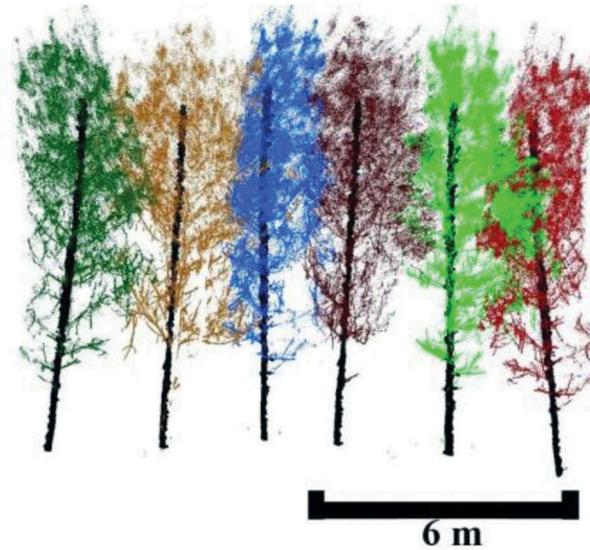
reference in the evaluation is obtained manually. The evaluation on the point level includes the completeness and correctness as calculated in Equation (4).

$$\text{Completeness} = \frac{TP}{TP + FN}, \text{Correctness} = \frac{TP}{TP + FP} \quad (4)$$

The comparison of our method and other methods is shown in Table 2.

Table 2. Comparisons of the completeness and correctness of different methods.

Method	Completeness (%)	Correctness (%)
Hetti Arachchige (2013)	92.8	97.5
Lehtomäki et al. (2010)	83.5	86.5
Xia et al. (2015)	88.0	93.0
Proposed	94.2	95.7

**Figure 10.** Illustration of the individual tree segmentation.

Results from [Table 2](#) indicate that MLS data are very promising information for the analysis of residential tree inventories. Our method obtains a high accuracy in both completeness and correctness which is proved to work well in the stem detection. It took us 2.5 minutes to detect 120/129 stems from the input 190,607,045 MLS points. Experiments were done on a Windows 10 Home 64-bit, Intel Core i5-7200U 2.5 GHz processor with 16 GB of RAM and computations were carried on Matlab R2017a. Moreover, our method also achieves a high accuracy in the individual tree segmentation when using the shortest-path algorithm (CSP) (West, Brown, and Enquist 1999) as shown in [Figure 10](#).

3. Conclusions

This letter proposes a method for the stem detection from residential environments using MLS data. Ground points are filtered by our point removal method based on the elevation histogram adaptively. Seeds of trees for the stem growing are chosen by the circle fitting from the 2D projection of points. MLS points are organized by adjacent voxels and then modeled in a graph for the stem optimization. In the optimization, the cost of building a stem is calculated by a formulated energy function and minimized by the dynamic programming. In the evaluation, our algorithm achieves an accuracy of 93.02% on the stem level, and a completeness of 94.2% and correctness of 95.7% on the point level. Our work shows that stems in residential environments can be localized and detected well

based on the proposed optimization method with MLS data. The stem detection in mountains where seeds are difficult to be set is required to be further studied.

Funding

This work was supported by National Key R&D Program of China (2017YFD06009), National Natural Science Foundation of China (31770591, 41701510), China Postdoctoral Science Foundation (2016M601823).

References

- Axelsson, P. 2000. "DEM Generation from Laser Scanner Data Using Adaptive TIN Models." *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences* XXXIII (Pt. B4/1): 110–117.
- Guan, H., Y. Yu, Z. Ji, J. Li, and Q. Zhang. 2015. "Deep Learning-Based Tree Classification Using Mobile LiDAR Data." *Remote Sensing Letters* 6 (11): 864–873. doi:10.1080/2150704X.2015.1088668.
- Hetti Arachchige, N. 2013. "Automatic Tree Stem Detection – A Geometric Feature Based Approach for MLS Point Clouds." *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences* 1 (2): 109–114. doi:10.5194/isprsannals-II-5-W2-109-2013.
- Lehtomäki, M., A. Jaakkola, J. Hyypä, A. Kukko, and H. Kaartinen. 2010. "Detection of Vertical Pole-Like Objects in a Road Environment Using Vehicle-Based Laser Scanning Data." *Remote Sensing* 2 (3): 641–664. doi:10.3390/rs2030641.
- Liang, X., J. Hyypä, A. Kukko, H. Kaartinen, A. Jaakkola, and X. Yu. 2014. "The Use of a Mobile Laser Scanning System for Mapping Large Forest Plots." *IEEE Geoscience and Remote Sensing Letters* 11 (9): 1504–1508. doi:10.1109/LGRS.2013.2297418.
- Lu, X., Q. Guo, W. Li, and J. Flanagan. 2014. "A Bottom-Up Approach to Segment Individual Deciduous Trees Using Leaf-Off Lidar Point Cloud Data." *ISPRS Journal of Photogrammetry and Remote Sensing* 94: 1–12. doi:10.1016/j.isprsjprs.2014.03.014.
- Matsuki, T., N. Yokoya, and A. Iwasaki. 2015. "Hyperspectral Tree Species Classification of Japanese Complex Mixed Forest with the Aid of LiDAR Data." *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 8 (5): 2177–2187. doi:10.1109/JSTARS.2015.2417859.
- Schnabel, R., R. Wahl, and R. Klein. 2007. "Efficient RANSAC for Point-Cloud Shape Detection." *Computer Graphics Forum* 26 (2): 214–226. doi:10.1111/cgf.2007.26.issue-2.
- West, G. B., J. H. Brown, and B. J. Enquist. 1999. "A General Model for the Structure and Allometry of Plant Vascular Systems." *Nature* 400 (6745): 664–667. doi:10.1038/23251.
- Xia, S., C. Wang, F. Pan, X. Xi, H. Zeng, and H. Liu. 2015. "Detecting Stems in Dense and Homogeneous Forest Using Single-Scan TLS." *Forests* 6 (11): 3923–3945. doi:10.3390/f6113923.
- Yun, T., F. An, W. Li, Y. Sun, L. Cao, and L. Xue. 2016. "A Novel Approach for Retrieving Tree Leaf Area from Ground-Based LiDAR." *Remote Sensing* 8 (11): 942. doi:10.3390/rs8110942.
- Zhong, L., L. Cheng, H. Xu, Y. Wu, Y. Chen, and M. Li. 2017. "Segmentation of Individual Trees from TLS and MLS Data." *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 10 (2): 774–787. doi:10.1109/JSTARS.2016.2565519.