



Automatic extraction of trees by using multiple return properties of the lidar point cloud

Samed Ozdemir ^{*1}, Zeynep Akbulut ¹, Fevzi Karsli ², Hayrettin Acar ²

¹Gümüşhane University, Engineering and Natural Sciences Faculty, Department of Geomatics Engineering, Gümüşhane, Turkey

²Karadeniz Technical University, Engineering Faculty, Department of Geomatics Engineering, Trabzon, Turkey

Keywords

Automatic
Tree Extraction
LiDAR
Multiple Return

ABSTRACT

Airborne laser scanning has been a valuable tool for forestry applications since it began to be used commercially. Thanks to the high 3D resolution provided by the Light Detection and Ranging (LiDAR) point cloud, it has provided great convenience in complex 3D modeling processes needed for forestry applications such as forest inventory, forest management, determination of carbon stocks and the characterization of biodiversity. LiDAR data provides a new dimension in forestry applications with its high 3D resolution and multiple return characteristics. The extraction of woodland areas from the LiDAR point cloud has great importance for automating the determination of tree heights, species and stand frequency which will be used for generating canopy height models (CHM). In this study, woodland areas in the urban scene were automatically extracted by using the multiple return properties of the LiDAR point cloud. The proposed approach consists of three major steps namely pre-processing, parameter calculation and k-d tree search for trees which were implemented in MATLAB. In the first step, multiple return points have been identified from the LiDAR point cloud, which will be then used to determine possible tree locations. Then, by using Density-Based Spatial Clustering of Applications with Noise (DBSCAN) algorithm, neighborhood relations among the multi return points which were extracted from the initial point cloud data, were formed and a rule-based filter was applied by taking advantage of neighborhood relations. In addition, the initial point cloud was filtered with the Cloth Simulation Filtering (CSF) algorithm to separate ground and non-ground points where non-ground points used to extract trees. In the second step, non-vegetation points were removed by applying a threshold based on the change of curvature and planarity parameters, which are derived from the filtered non-ground point cloud. In the last step, in order to extract trees, a k-d tree structure was created from the filtered non-ground points to find nearest neighbors of each multi return point within a given diameter in the k-d tree structure. In order to evaluate the accuracy of the approach, the extracted boundaries were compared with the manually digitized woodland boundaries from the true orthophoto of the study area using correctness, completeness and quality metrics.

1. INTRODUCTION

Remote sensing data has become a reliable source for determining forest parameters due to the broad data coverage, data accessibility and advanced image processing techniques, both at the individual tree level and at the tree community level (Chen and Zhu 2013; Mielcarek et al. 2018). However, 2D systems do not provide important structural and geometric information such as the internal structure of the trees and canopy characteristics. Therefore,

there is a need for 3D information including detailed geometric and structural information about the trees (Liu et al. 2019; Ramiya et al. 2019). Nowadays, LiDAR data is actively being used in many different kinds of forestry applications thanks to the characteristic trait of LiDAR signals' penetration capability of tree canopies (Liu et al. 2013; Vêga et al. 2014). Penetration capability of LiDAR signals provides accurate information about the tree structures and the ground beneath the trees

* Corresponding Author

(samed.ozdemir@gumushane.edu.tr) ORCID ID 0000 - 0001 - 7217 - 899X
(zeynepakbulut@gumushane.edu.tr) ORCID ID 0000 - 0001 - 9801 - 1506
(fkarsli@ktu.edu.tr) ORCID ID 0000 - 0002 - 0411 - 3315
(h_acar@ktu.edu.tr) ORCID ID 0000-0002-2954-7734

Research Article / DOI: 10.26833/ijeg.668352

Cite this article

Ozdemir S, Akbulut Z, Karsli F & Acar H (2021). Automatic extraction of trees by using multiple return properties of the lidar point cloud. International Journal of Engineering and Geosciences, 6(1), 20-26

(Reutebuch et al. 2003; Vége et al. 2014; Mielcarek et al. 2018; Liu et al. 2019; Ramiya et al. 2019). This property makes LiDAR a powerful tool for monitoring, assessment and segmentation of forest areas, tree canopies and individual trees due to its speed, coverage and 3D information (Vége et al. 2014; Beland et al. 2019; Liu et al. 2019; Ramiya et al. 2019).

Mapping and monitoring of urban woodland areas have vital importance due to the effects of urban woodland areas on local micro-climate, air quality and ecosystem. It is also a known fact that urban green spaces improve health, wellbeing, and quality of life for urban residents (Haq 2011; Gupta et al. 2018; Ramiya et al. 2019). Successful results have been obtained in many studies using LiDAR point cloud data for forestry applications. Widely used methods for forestry applications can be categorized into three groups, which are raster-based, point cloud based and hybrid methods. Raster-based methods work by locating each canopy over the local maxima via the canopy height model (CHM) (Popescu 2007). Point-based methods allow the determination of each canopy by using local geometry and structural features amongst the points within a local neighborhood (Zhen et al. 2016; Ramiya et al. 2019). There are a few things to consider here. Raster-based methods reduce the complexity of computations and increase performance while losing the geometric and structural information from the point cloud. On the other hand, although point-based methods are computationally expensive, they retain the structural and geometric properties of the canopy and make the most of this information. In the hybrid method, canopies are located from 2D CHMs by using local maxima based on image processing techniques and tree points are extracted from the point cloud using this location information (Lindberg et al. 2014; Liu et al. 2019).

In the last two decades LiDAR technology used for measuring the tree canopy by using both terrestrial and airborne LiDAR data (Hyyppä et al. 2001; Van der Zande et al. 2006; Koch et al. 2006). Moreover, new methodologies developed by merging LiDAR point cloud data with remotely sensed images to extract trees (Dogon-Yaro et al. 2016; Hartling et al. 2019). However, most of the studies focused on trees in forested areas (Liu et al. 2013). Trees located in an urban environment also as important as trees in forested areas because they are closely related to the residents of the urban environment (Liu et al. 2013). Secord and Zakhor (2007) proposed an approach for automatic detection of trees using LiDAR and aerial imagery using Support Vector Machines (SVM). Despite obtaining good results, collecting a huge number of training data required to train SVM algorithm is not suitable for most applications. Besides, aerial imagery must be precisely registered with the LiDAR data to obtain the desired accuracy. Liu et al. (2013) used only LiDAR data to extract individual tree

crowns in urban areas by using multiple return properties to segment trees with a surface growing algorithm. The proposed algorithm extracted 85% of the trees located in the test areas. Gupta et al. (2018) used multi return information of LiDAR point cloud to identify individual tree canopies by voxelizing the point cloud data and applying a connected component analysis. The proposed method achieved approximately 90% percent accuracy in a high-resolution urban LiDAR dataset. Ramiya et al. (2019) used supervoxels and geometric information of LiDAR point cloud such as pointedness, curviness and surfacedness parameters to classify individual tree crowns using random forest classification. Their approach achieved 99% accuracy with high-resolution LiDAR point cloud data in an urban environment.

In this paper, a new approach proposed to extract tree canopies by using multiple return properties of LiDAR data in urban areas. The proposed approach consists of three steps that were implemented in MATLAB. In the first step, the LiDAR point cloud was filtered with the CSF algorithm to detect ground and non-ground points. Then, the multi return points were extracted from the point cloud and clustered with the DBSCAN algorithm to create neighborhood relations and filter out possible outlier points. In the second step, change of curvature and planarity parameters were calculated from the filtered non-ground points to distinguish trees from the non-vegetation objects such as buildings. In the last step, a k-d tree structure created from the remaining points, which were filtered with the previously mentioned parameters. Finally, a range search was initialized in the k-d tree structure by using the multiple return points and the tree points were extracted. Accuracy assessment of the proposed approach was evaluated with the correctness, completeness and quality metrics.

2. STUDY AREA and DATASET METHOD

The study areas were selected from the ISPRS's Vaihingen dataset (Cramer 2010) which includes LiDAR point cloud and an orthophoto of the region. In the benchmark dataset, LiDAR data has an average density of 8 points/m² and up to four returns recorded for each signal. The orthophoto has 9 cm ground sampling distance and three channels (NIR, green and blue). The study areas that are shown in (Figure 1) comprises of trees with varying density, size, shape and height. Also, some single and small trees can be found along with landscaping for both of the study areas. Study areas specifically selected for study area 1, there are three multi-story buildings with heights approximately 20 meters and none of the building's roofs obstructed by trees. For study area 2, all of the buildings are detached and some building's roofs are partially obstructed by a nearby tree. The study areas were chosen considering the complex relationships of buildings and trees in an urban scene.

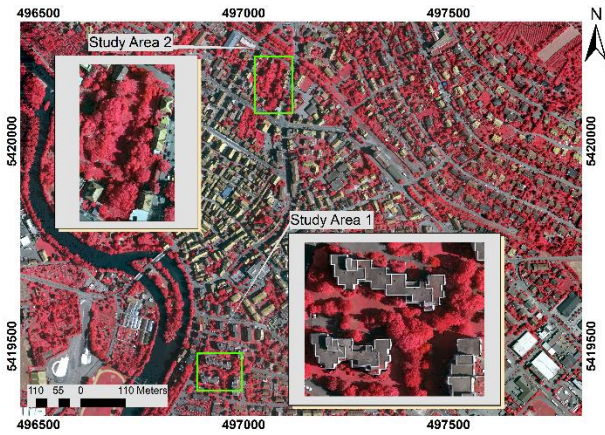


Figure 1. Study areas

3. METHODOLOGY

In order to extract trees, a three steps methodology namely; pre-processing, parameter calculation and k-d tree search for trees were implemented in MATLAB. All of the steps are explained in detail in the following sections.

3.1 Pre-Processing

Pre-processing contains clustering and filtering steps, which are described in detail in the sub-sections.

3.1.1 Clustering

Firstly, multiple return points were extracted from the initial LiDAR point cloud, which were later used to search for trees. These points were then separated into different clusters with the DBSCAN algorithm according to the maximum distance and minimum neighboring point parameters. The maximum distance parameter determines if a point is close to any other points in a cluster. In addition, the minimum neighboring point's parameter determines if the cluster has enough points to be considered as a cluster. If a cluster fulfills these two conditions, they will be labeled with a cluster number and the DBSCAN algorithm seeks for other clusters in the remaining points (Ester et al. 1996). The maximum distance and minimum neighbor number parameters were selected as 3 m and 5 points, respectively. Points that meet these conditions clustered together and labeled with a cluster number otherwise points marked as noise and were removed from the multi return point cloud.

3.1.2 Filtering

In order to avoid errors that may arise from ground points and the points close to the ground such as LiDAR returns from vehicles and low vegetation, point cloud must be filtered with an appropriate filtering algorithm. In this study, CSF algorithm (Zhang et al. 2016) was used for LiDAR point cloud filtering. CSF algorithm based on the simulation of a cloth model. Cloth simulation is a

term used in 3D computer graphics. The cloth can be modeled as a grid structure in which every grid node has mass and interconnections. The cloth model's grid nodes' interconnections are modeled as virtual springs which are governed by Hooke's law. Several modifications have been made to the algorithm to make the simulation run on LiDAR point clouds. First, the movement of grid nodes constrained in the vertical direction to detect collisions by comparing height values of grid node and terrain. Second, when a node reaches the ground, this node is set as unmovable. Third, cloth model simulation simplified by omitting collision forces operating on grid nodes (Zhang et al. 2016).

To detect ground points, firstly, CSF method turns point cloud upside down and fits a cloth model to this point cloud with the given cloth parameters as shown in Figure 2. Then, the cloth model's nodes interact with the corresponding points in the point cloud to find a suitable location. After the cloth simulation ends, ground points can be detected with the final shape of the cloth model by measuring distances between the cloth model and points (Figure 2).

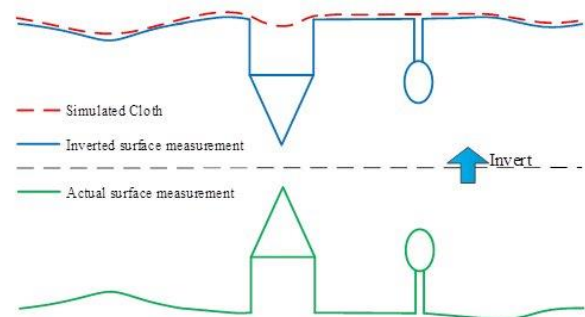


Figure 2. Overview of the CSF algorithm (Zhang et al. 2016)

Ground and non-ground point classes were obtained with the CSF algorithm. After the filtering process, non-ground points still have low-lying points (vehicles, low vegetation, etc.) which should be cleared to improve the final result of the study. Therefore, a Digital Terrain Model (DTM) was generated from the ground points and this surface elevated by two meters. Then, the obtained surface was applied to the non-ground points to clear the points under the surface. Thus, low objects such as vehicles and low vegetation were removed from the non-ground points (Figure 3).

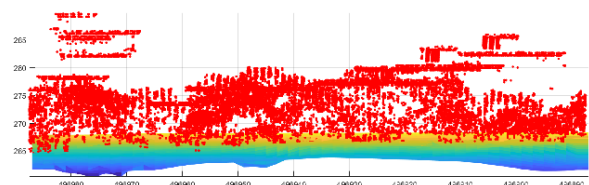


Figure 3. First study area: non-ground LiDAR points shown in red above the elevated ground surface

3.2 Parameter Calculation

Depending on the flight path and scan angle of the LiDAR instrument, trees and building edges can cause multi return points. Multi return LiDAR points from the buildings must be eliminated in order to search the LiDAR point cloud for neighbors of multi return points. To remove the multi return points from the buildings, we used planarity (Eq. 1) and change of curvature (Eq. 2) parameters (Pauly et al. 2003; Weinmann et al. 2015; Sevgen 2018) which was calculated from the filtered non-ground points (Figure 4). These eigenvalues contain valuable information about the local surface variation. 3D structure tensor of 3D covariance matrix of the 3D point cloud represents a symmetric positive semidefinite matrix which has three nonnegative eigenvalues and corresponding eigenvectors. Therefore, this information can be used to acquire local features around a 3D point (Weinmann 2016).

$$\text{Planarity} = \frac{e_2 - e_3}{e_1} \quad (1)$$

$$\text{Change of Curvature} = \frac{e_3}{\sum_{i=1}^3 e_i} \quad (2)$$

Where e_i denotes the eigenvalues of the covariance matrix of neighboring points with the subscript numbers representing the first, second and third eigenvalues in $e_1 \geq e_2 \geq e_3$ order.

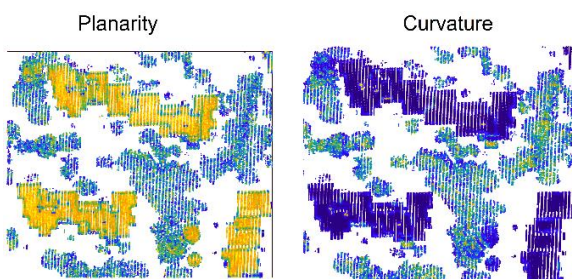


Figure 4. Calculated planarity and change of curvature parameters for study area 1. Blue color represents the value of 0 and yellow color represents the value of 1 for the calculated parameters.

A simple threshold-based method was employed in order to detect and remove the multi return building points from the point cloud using planarity and change of curvature parameters. Firstly, the change of curvature and planarity parameters were normalized between 0 to 1 value. A new parameter was calculated dividing planarity by the change of curvature parameter which was then used for thresholding. The threshold value was determined by Otsu's threshold method proposed by Otsu N. (1979) which selects a threshold value to maximize the variance between high and low parameter values. Points having a higher value from the calculated threshold and their neighboring multi-return points within a 1 m search radius were

removed from the point cloud. Detected multi-return points belonging to possible trees are shown in Figure 5.

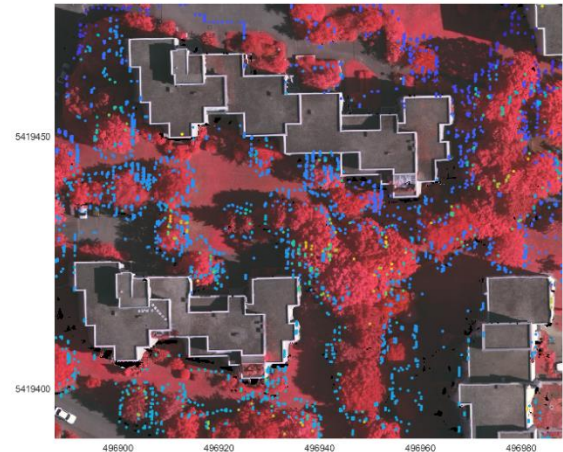


Figure 5. Possible multi-return tree points for the study area 1

3.3 K-D Tree Search for Tree Points

In the point cloud, which was filtered according to the change of curvature and planarity parameters; it was ensured that the neighborhoods of multi return points can be found quickly using the k-d tree structure. The K-d tree structure is a widely used method for database searches, statistics and computer vision. The K-d tree method is a binary tree method developed by Bentley, J. L. (1975). This method allows quick search within a large data set by splitting a given data set into many sub-segments with hyperplanes. These sub-segments are called leaf nodes which are indicated with a pointer. For a given point, K-d tree method quickly searches the data set using pointers to identify the leaf node closest to the given point. In the tree structure, all the neighbors with a diameter of 5 meters were determined using the multi return points produced in the data preprocessing stage. Points with less than or equal to 3 points in the neighborhood were discarded and the remaining points were recorded as tree points is shown in Figure 6 and Figure 7.

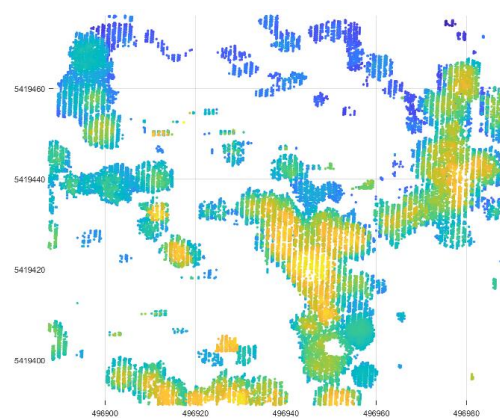


Figure 6. K-d tree search results for tree identification from multi-return points for study area 1

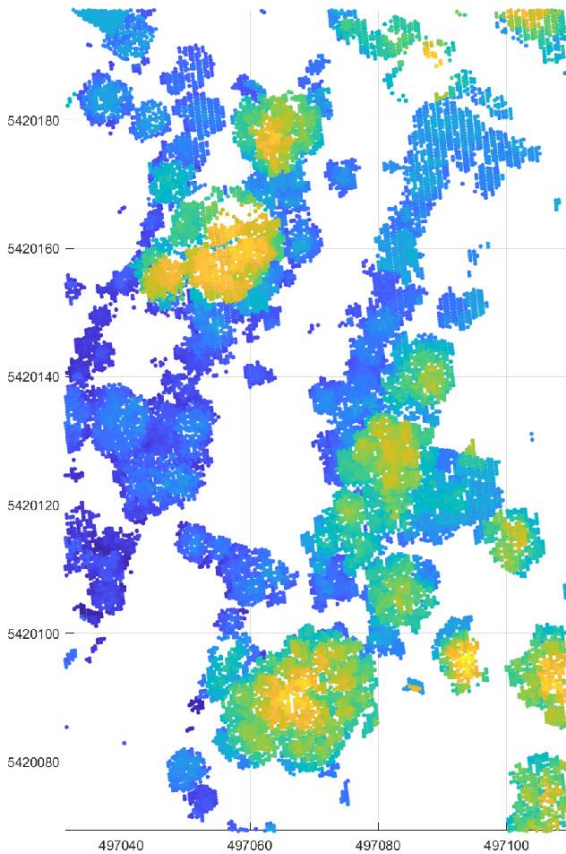


Figure 7. K-d tree search results for tree identification from multi-return points for study area 2

4. RESULTS and DISCUSSION

The proposed approach was evaluated in two urban study areas with various types of tree structures and stand types. Extracted tree boundaries were compared with the manually digitized reference tree boundaries. For the two study areas, accuracy assessment was evaluated with the correctness, completeness and quality metrics by the following equations (3, 4, 5). Accuracy assessment results are shown in Table 1.

$$\text{Completeness} = \frac{\|TP\|}{\|TP\| + \|FN\|} \quad (3)$$

$$\text{Correctness} = \frac{\|TP\|}{\|TP\| + \|FP\|} \quad (4)$$

$$\text{Quality} = \frac{\|TP\|}{\|TP\| + \|FN\| + \|FP\|} \quad (5)$$

Where, TP refers to an entity classified as an object that also corresponds to an object in the reference is classified as a true positive, FN (false negative) refers to an entity corresponds to an object in the reference that is classified as background, FP (false positive) refers to an entity classified as an object that does not correspond to an object in the reference and TN (true negative) refers to an entity belongs to the background both in the classification

and in the reference data (Rutzinger et al. 2009, Karsli et al. 2016, Akbulut et al. 2018). Reference data was created from the LiDAR point cloud by manually selecting tree points. Then, collected tree points were used to create a binary reference raster by mapping every point to a raster cell. After creating a reference raster, void pixels caused by point spacing were filled with a dilation operation using a circular structuring element.

Table 1. Accuracy assessment results for study areas

	Correctness	Completeness	Quality
Area 1	0.9157	0.8794	0.8135
Area 2	0.9456	0.8569	0.8167

For both of the two test areas, the proposed approach extracted trees with reasonable accuracy. Especially, large tree canopies consisting of multiple trees and single trees with relatively wide canopy and height, extracted with reasonable accuracy as shown in Figure 8.

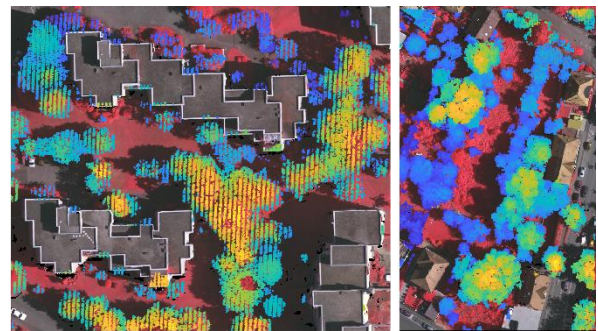


Figure 8. Tree extraction results were obtained with the proposed approach. The left image shows tree extraction results for study area 1, and the right image shows tree extraction results for study area 2.

However, single trees with low height and thin canopies were not extracted because most of the above ground objects lower than two meters were filtered. Also, some of the building roofs intertwined with the trees were complicated the tree extraction process, which can be seen in results for study area 2 in Figure.8. Moreover, high buildings have LiDAR returns from the building's walls and this situation creates some problems for the parameter calculation step that results in misidentified tree points. Overall, the proposed approach was achieved over 90% correctness, 85% completeness, and 81% quality, which can be improved in later studies. The accuracy assessment results acquired with the proposed approach compared with the results from Liu et al. (2013) and Gupta et al. (2018). Liu et al. (2013) assessed the accuracy of their approach on two test areas and acquired 92% and 94% correctness, 87% and 85% completeness. Gupta et al. (2018) acquired 88% correctness and 89% completeness for a single test area. In the light of these comparisons, the proposed approach performed reasonably well with some minor flaws.

5. CONCLUSION

Promising results have been achieved in the Vaihingen data set with the proposed approach. The dense tree clusters with multiple trees were determined with reasonably high accuracy. The determination of individual trees was achieved with an appreciable success considering the canopy structure and height of the tree. The proposed approach may give different results when the trees and buildings are adjacent to each other. The reasons for this are that the parameters obtained from the point cloud depend on the changes in the neighborhood distance and LiDAR returns from the building walls. The proposed approach will be improved by addressing the aforementioned problems in future studies.

REFERENCES

- Akbulut Z, Özdemir S, Acar H, Dihkan M & Karlı F (2018). Automatic extraction of building boundaries from high-resolution images with active contour segmentation. *International Journal of Engineering and Geosciences*, 3(1), 36-42.
- Beland M, Parker G, Sparrow B, Harding D, Chasmer L, Phinn S, Antonarakis A & Strahler A (2019). On promoting the use of lidar systems in forest ecosystem research. *Forest Ecology and Management*, 450, 117484. DOI: 10.1016/j.foreco.2019.117484
- Bentley J L (1975). Multidimensional binary search trees used for associative searching. *Communications of the ACM*, 18(9), 509-517. DOI: 10.1145/361002.361007
- Chen Y & Zhu X (2013). An integrated GIS tool for automatic forest inventory estimates of *Pinus radiata* from LiDAR data. *GIScience & remote sensing*, 50(6), 667-689. DOI: 10.1080/15481603.2013.866783
- Cramer M (2010). The DGPf-test on digital airborne camera evaluation-overview and test design. *Photogrammetrie-Fernerkundung-Geoinformation*, 2010(2), 73-82. DOI: 10.1127/1432-8364/2010/0041
- Dogon-Yaro M A, Kumar P, Rahman A A & Buyuksalih G (2016). Extraction of urban trees from integrated airborne based digital image and LiDAR point cloud datasets-initial results. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 42, 81.
- Ester M, Kriegel H P, Sander J & Xu X (1996, August). A density-based algorithm for discovering clusters in large spatial databases with noise. In *Kdd*, 96(34), 226-231.
- Gupta A, Byrne J, Moloney D, Watson S & Yin H (2018). Automatic Tree Annotation in LiDAR Data. In *Proceedings of the 4th International Conference on Geographical Information Systems Theory, Applications and Management*, 36-41.
- Haq S M A (2011). Urban green spaces and an integrative approach to sustainable environment. *Journal of Environmental Protection*, 2(05), 601-608.
- Hartling S, Sagan V, Sidike P, Maimaitijiang M & Carron J (2019). Urban Tree Species Classification Using a WorldView-2/3 and LiDAR Data Fusion Approach and Deep Learning. *Sensors*, 19(6), 1284. DOI: 10.3390/s19061284
- Hyypä, J, Schardt M, Haggrén H, Koch B, Lohr U, Scherrer H U et al. (2001). HIGH-SCAN: The first European-wide attempt to derive single-tree information from laser scanner data. *The Photogrammetric Journal of Finland*, 17(2), 58-68.
- Karsli F, Dihkan M, Acar H & Ozturk A (2016). Automatic building extraction from very high-resolution image and LiDAR data with SVM algorithm. *Arabian Journal of Geosciences*, 9(14), 635. DOI: 10.1007/s12517-016-2664-7
- Koch B, Heyder U & Weinacker H (2006). Detection of individual tree crowns in airborne lidar data. *Photogrammetric Engineering & Remote Sensing*, 72(4), 357-363.
- Lindberg E, Eysn L, Hollaus M, Holmgren J & Pfeifer N (2014). Delineation of tree crowns and tree species classification from full-waveform airborne laser scanning data using 3-D ellipsoidal clustering. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 7(7), 3174-3181.
- Liu J, Shen J, Zhao R & Xu S (2013). Extraction of individual tree crowns from airborne LiDAR data in human settlements. *Mathematical and Computer Modelling*, 58(3-4), 524-535.
- Liu L, Lim S, Shen X & Yebra M (2019). A hybrid method for segmenting individual trees from airborne lidar data. *Computers and Electronics in Agriculture*, 163, 104871.
- Mielcarek M, Stereńczak K & Khosravipour A (2018). Testing and evaluating different LiDAR-derived canopy height model generation methods for tree height estimation. *International Journal of Applied Earth Observation and Geoinformation*, 71, 132-143.
- Otsu N (1979). A threshold selection method from gray-level histograms. *IEEE transactions on systems, man, and cybernetics*, 9(1), 62-66.
- Pauly M, Keiser R & Gross M (2003). Multi-scale feature extraction on point-sampled surfaces. In *Computer graphics forum*, 22(3), 281-289.
- Popescu S C (2007). Estimating biomass of individual pine trees using airborne lidar. *Biomass and Bioenergy*, 31(9), 646-655.
- Ramiya A M, Nidamanuri R R & Krishnan R (2019). Individual tree detection from airborne laser scanning data based on supervoxels and local convexity. *Remote Sensing Applications: Society and Environment*, 15, 100242.

- Reutebuch S E, McGaughey R J, Andersen H E & Carson W W (2003). Accuracy of a high-resolution lidar terrain model under a conifer forest canopy. *Canadian Journal of Remote Sensing*, 29(5), 527-535.
- Rutzinger M, Rottensteiner F & Pfeifer N (2009). A comparison of evaluation techniques for building extraction from airborne laser scanning. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 2(1), 11-20.
- Secord J & Zakhor A (2007). Tree detection in urban regions using aerial LiDAR and image data. *IEEE Geoscience and Remote Sensing Letters*, 4(2), 196-200.
- Sevgen S C (2018). Airborne lidar data classification in complex urban area using random forest: A case study of Bergama, Turkey. *International Journal of Engineering and Geosciences*, 4(1), 45-51. DOI: 10.26833/ijeg.440828
- Van der Zande D, Hoet W, Jonckheere I, van Aardt J & Coppin P (2006). Influence of measurement set-up of ground-based LiDAR for derivation of tree structure. *Agricultural and Forest Meteorology*, 141(2-4), 147-160. DOI: 10.1016/j.agrformet.2006.09.007
- Véga C, Hamrouni A, El Mokhtari S, Morel J, Bock J, Renaud J P & Durrieu S (2014). PTrees: A point-based approach to forest tree extraction from LiDAR data. *International Journal of Applied Earth Observation and Geoinformation*, 33, 98-108. DOI: 10.1016/j.jag.2014.05.001
- Weinmann M (2016). *Reconstruction and analysis of 3D scenes*. Springer.
- Weinmann M, Mallet C, Hinz S & Jutzi B (2015). Efficient interpretation of 3D point clouds by assessing feature relevance. *AVN–Allg Vermessung-Nachr*, 10(2015), 308-315.
- Zhang W, Qi J, Wan P, Wang H, Xie D, Wang X & Yan G (2016). An easy-to-use airborne LiDAR data filtering method based on cloth simulation. *Remote Sensing*, 8(6), 501. DOI: 10.3390/rs8060501
- Zhen Z, Quackenbush L J & Zhang L (2016). Trends in automatic individual tree crown detection and delineation—Evolution of LiDAR data. *Remote Sensing*, 8(4), 333. DOI: 10.3390/rs8040333



© Author(s) 2021.

This work is distributed under <https://creativecommons.org/licenses/by-sa/4.0/>