AUTOMATIC EXTRACTION OF TREES BY USING THE MULTIPLE RETURN PROPERTIES OF THE LIDAR POINT CLOUD

S. Özdemir [a, *], Z. Akbulut [a], F. Karslı [b], H. Acar [b]

[a] Gumushane University, Engineering and Natural Sciences Faculty, Geomatics Engineering Department, 29000 Gümüşhane, Turkey – {samed.ozdemir, zeynepakbulut}@gumushane.edu.tr

[b] Karadeniz Technical University, Engineering Faculty, Geomatics Engineering Department, 61080 Trabzon, Turkey – {fkarsli, h_acar}@ktu.edu.tr

KEY WORDS: 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 LiDAR point cloud, it has provided great convenience in complex 3D modelling 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. Extraction of woodland areas from the LiDAR point cloud have a 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. Proposed approach consists of three major steps that 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, neighbourhood 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 neighbourhood relations. In addition, the initial point cloud 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 removed by applying a threshold based on 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 neighbours 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

Nowadays, Light Detection and Ranging (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 (Vega et al., 2014). Penetration capability of LiDAR signals provides accurate information about the tree structures and the ground beneath the trees (Reutebuch et al., 2003, Vega et al., 2014). This property makes LiDAR a powerful tool for monitoring, assessment and segmentation of forest areas, tree canopies and individual trees (Vega et al., 2014).

Dense 3D point cloud data provided by LiDAR opened new possibilities to mathematically describe a tree's 3D structure for modelling the tree canopy (Liu et al., 2013). In the recent years LiDAR technology used for measuring the tree canopy by using both terrestrial an airborne LiDAR data (Hyyppa et al., 2001, 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 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 with the habitants 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. In addition, aerial imagery must be precisely registered with the LiDAR data to obtain 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. Proposed algorithm's extracted 85% of the trees located in the test areas.

In this paper, a new approach proposed to extract tree canopies by using multiple return properties of LiDAR data. Proposed approach consists of three steps that were implemented in Matlab. In the first step, LiDAR point cloud filtered with the CSF algorithm to detect ground and non-ground points. Then, the multi return points extracted from the point cloud and clustered with the DBSCAN algorithm to create neighbourhood relations and filter out possible outlier points. In the second step, 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. Then, a range search was initialized in the k-d tree structure by using the multiple return points and the tree points were extracted.

* Corresponding author.

2. STUDY AREA AND DATASET

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/m^2 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 is 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.

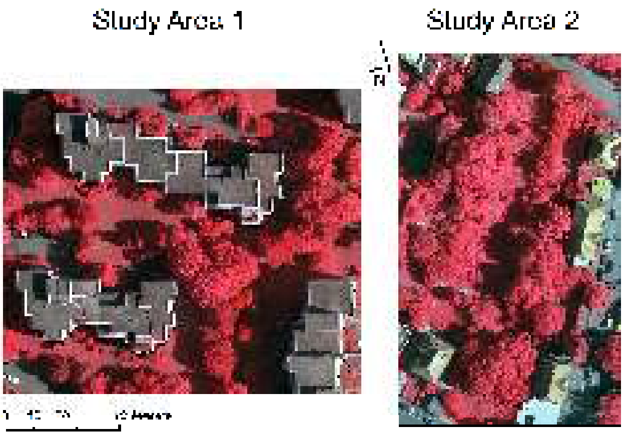

Figure 1. The study areas

3. METHODOLOGY

In order to extract trees, a three step 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

3.1.1 Clustering: In the pre-processing step, firstly multiple return points was 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 neighbouring point parameters. Maximum distance parameter determines if a point is close to any other points in a cluster. In addition, minimum neighbouring point's parameter determines if the cluster has enough points to be considered as a cluster. If a cluster fulfils these two conditions, they will be given a cluster number and the DBSCAN algorithm seeks for other clusters in the remaining points (Ester et al., 1996). The maximum distance and minimum neighbour number parameters were selected as 3 m and 5 points, respectively. Points satisfy these conditions clustered together and given a cluster number, otherwise points marked as noise and removed from the multi return point cloud.

3.1.2 Filtering: In order to prevent errors that may arise from ground points and the points close to 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 first turns point cloud upside down and fits a cloth model to this point cloud with the given cloth parameter. Then this cloth model's nodes interact with the corresponding points to find a suitable location. Moreover, ground points can be filtered with the final shape of the cloth model (Figure 2).

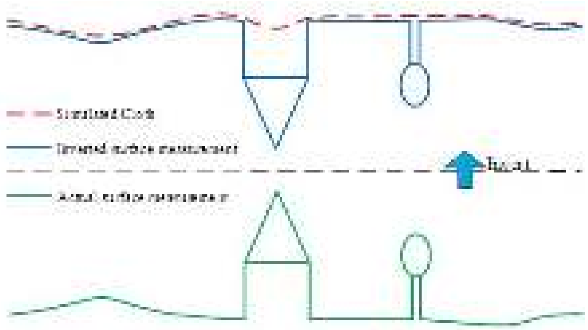

Figure 2. Overview of the CSF algorithm (Zhang et al. 2016)

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 must be cleared to improve the final result of the study. Thus, a Digital Terrain Model (DTM) was generated from the ground points and this surface elevated by two meters. Then, the obtained surface applied to the non-ground points to clear the points under this surface. Thus, vehicles and low objects were removed from the non-ground points.

3.2 Parameter Calculation

As a characteristic of LiDAR, multi return points can be caused by trees and building edges. In order to search the LiDAR point cloud for neighbours of multi return points, LiDAR returns from the buildings must be eliminated. To remove the LiDAR points returned from the buildings, curvature (Eqn. 1) and planarity (Eqn. 2) parameters (Pauly et al., 2003; Weinmann et al., 2015) used which was calculated from the filtered non-ground points (Figure 3).

$$Planarity = \frac{e_2 - e_3}{e_1} \tag{1}$$

$$Curvature = \frac{e_3}{\sum_{i=1}^{3} e_i} \tag{2}$$

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

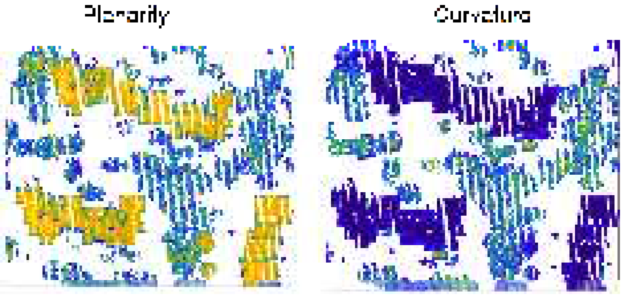

Figure 3. Calculated planarity and curvature parameters for study area 1. Blue colour represents value of 0 and yellow colour represents value of 1 for the calculated parameters.

3.3 K-D Tree Search for Tree Points

In the point cloud, which was filtered according to the curvature and planarity parameters; it was ensured that the neighbourhoods of multi return points can be found quickly by 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 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 by 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.

4. RESULTS AND DISCUSSION

Proposed approach was evaluated on 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 all test areas, accuracy assessment was carried out with the correctness, completeness and quality metrics by the following equations (3, 4, 5). Accuracy assessment results are shown in Table 1.

$$Completeness=\frac{\|TP\|}{\|TP\|+\|FN\|} \tag{3}$$

$$Correctness=\frac{\|TP\|}{\|TP\|+\|FP\|} \tag{4}$$

$$Quality=\frac{\|TP\|}{\|TP\|+\|FN\|+\|FP\|} \tag{5}$$

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 corresponds 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).

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

Table 1. Accuracy assessment results

For both of the two test areas, proposed approach successfully extracted trees. 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 4. However, single trees with low height and thin canopies was not extracted because most of the above ground objects lower than two meters was filtered. Also, some of the building roofs intertwined with the trees which was complicated the tree extraction process, which can be seen in results for study area 2 in Figure 4. Moreover, high buildings have LiDAR returns from the building's walls and this situation creates problems for the parameter calculation step that results in misidentified tree points. Overall, proposed approach was achieved over 90% correctness, 85% completeness and 81% quality, which can be improved in the later studies. The accuracy assessment results acquired with the proposed approach compared with Liu et al., (2013) and Gupta et al., (2018) accuracy assessment results. 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 comparison, proposed approach performed as expected with some minor flaws.

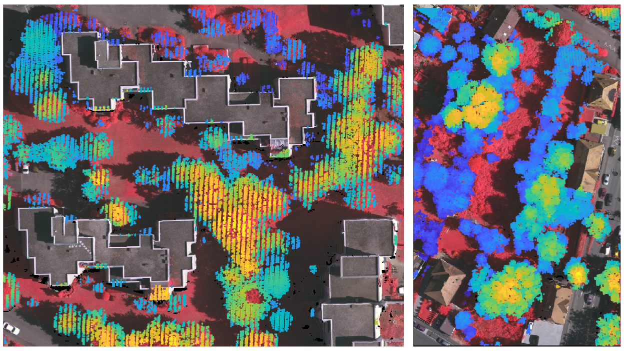

Figure 4. Tree extraction results were obtained with the proposed approach. Left image shows tree extraction results for study area 1, and right image shows tree extraction results for study area 2.

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 high accuracy. The determination of individual trees were 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 reason for this is that the parameters obtained from the point cloud depend on the changes in the neighborhood distance and LiDAR returns from the building walls. Proposed approach will be improved by addressing aforementioned problems in the future studies.

REFERENCES

Bentley, J. L. (1975). Multidimensional binary search trees used for associative searching. Communications of the ACM, 18(9), 509-517.

Cramer, M. (2010). The DGPF-test on digital airborne camera evaluation–overview and test design. Photogrammetrie-Fernerkundung-Geoinformation, 2010(2), 73-82.

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 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 (Vol. 96, No. 34, pp. 226-231).

Gupta, A., Byrne, J., Moloney, D., Watson, S., & Yin, H. (2018). Automatic Tree Annotation in LiDAR Data. In GISTAM (pp. 36-41).

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.

Hyyppä, J., Schardt, M., Haggrén, H., Koch, B., Lohr, U., Scherrer, H. U., & Pyysalo, U. (2001). HIGH-SCAN: The first European-wide attempt to derive single-tree information from laserscanner 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.

Koch, B., Heyder, U., & Weinacker, H. (2006). Detection of individual tree crowns in airborne lidar data. Photogrammetric Engineering & Remote Sensing, 72(4), 357-363.

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.

Pauly, M., Keiser, R., & Gross, M. (2003, September). Multi-scale feature extraction on point-sampled surfaces. In Computer graphics forum (Vol. 22, No. 3, pp. 281-289). Oxford, UK: Blackwell Publishing, Inc.

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.

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.

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.

Weinmann, M., Mallet, C., Hinz, S., & Jutzi, B. (2015). Efficient interpretation of 3D point clouds by assessing feature relevance. AVN–Allg Vermess-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.