Semantic Segmentation of Sparse 3D Point Cloud Based on Geometrical Features for Trellis-Structured Apple Orchard

Lihua Zeng[1,2,#], Juan Feng[1,3,#], Long He[1*]

[1]Department of Agricultural and Biological Engineering, The Pennsylvania State University, University Park, PA 16802, USA

[2]College of Mechanical and Electrical Engineering, Hebei Agricultural University, Baoding 071001, China

[3]College of Information Science and Technology, Hebei Agricultural University, Baoding 071001, China

[#] Equally contributed to the work

[*]**Corresponding author**: luh378@psu.edu

Abstract

Orchard operations such as mechanical pruning and spraying are heavily affected by tree architectures. Quantified inputs (e.g., cutting locations for mechanical pruning, and canopy distribution and density for variable-rate precision spraying) are necessary information for achieving precise control of these orchard operations. Even in planar orchard systems, trees grow differently. Therefore, it is essential to measure the canopy at the individual tree level. A three-dimensional (3D) light detection and ranging (LiDAR) sensor imaging system was developed to estimate the main canopy specifications. The LiDAR sensor was installed on a utility vehicle and driven alongside tree rows in an apple orchard. A total of 1,138 frames of point cloud data were acquired from 69 apple trees in a tall spindle architecture. An algorithm was developed in the MATLAB environment to segment trellis wires, support poles, and tree trunks in these point cloud images. The results indicated that the proposed algorithm achieved overall accuracy values of 88.6%, 82.1%, and 94.7%, respectively, in identifying the corresponding three objects. Furthermore, canopy density and depth maps were created with the

distribution of points in the point cloud images. The outcomes from this study provide baseline information for precision orchard operations such as mechanical pruning and precision spraying.

Keywords: apple trees; canopy density; point cloud; semantic segmentation; three-dimensional LiDAR

Nomenclature

boundary()	a function for finding the boundary of a cluster of points
Density	density of the point cloud data
Depth	depth of the tree canopy (m)
Dbscan()	a function for identifying high-density areas
Depth_{av}	average depth of a canopy area (m)
epsilon	neighbourhood search radius around a point
H	height of the tree canopy (m)
labels	cluster index of high-density point areas
LiDAR	Light Detection and Ranging
Min(y)	minimum *y*-axis value for the canopy points (m)
Max(z)	maximum *z*-axis value for the canopy points (m)
Min(z)	minimum *z*-axis value for the canopy points (m)
MSAC	M-estimator SAmple Consensus
minpts	required minimal neighbouring points for a core point
N	number of points in a fixed region
ROI	region of interest
RANSAC	RANdom SAmple Consensus
S	area of the fixed region (m^2)
X	input point set
$y_{\mathrm{trellisplane}}$	*y*-axis value of the trellis plane (m)
Ypmin-δ	lower limit of the *y*-axis value (m)
Ypmax+δ	upper limit of the *y*-axis value (m)

1. Introduction

Geometric and structural parameters are important for identifying the volume and/or shape of fruit trees for various production operations, especially precision applications. For example, the height, width, depth, and density of a tree crown are used as guide for variable-rate spraying

(Li et al., 2018; Wandkar, Bhatt, Jain, Nalawade, Pawar, 2018), biomass estimating (Fernández-Sarríaa, López-Cortés, Estornell, Velázquez-Martí, Salazar, 2019), automation machine application (He & Schupp, 2018), and plant evaluation and management (Colaço, Molin, Rosell-Polo, Escolà, 2018; Tagarakis, Koundouras, Fountas, Gemtos, 2018). More and more newly planted apple orchards are being trained to high-density tree structures, significantly improving yield and fruit quality (Lordan, Francescatto, Dominguez, Robinson, 2018).

In a typical high-density commercial apple orchard, a trellis support system with metal wires and steel or wooden poles is used to incur certain advantages. In particular, such a system may be used to encourage trees to direct energy to fruiting, provide a structural framework for tree training, improve light interception, produce earlier yields, or reduce labor costs (Ontario Apple Grower, 2015). The trellis system also can be used as an identified object for automated operations; for example, knowing the location of the trellis plane is helpful for orchard platform navigation, mechanical pruning, and spraying (Palleja Cabre, Llorens, Landers, 2017). Trellis wires, however, are usually too thin to be easily detected. Take pruning as an example: Regardless of whether pruning is done by humans or a machine, it is necessary to identify and avoid the trellis wires. Therefore, the identification and segmentation of trellis wires and poles are important tasks for an orchard, especially in relation to mechanical and automated operations.

A variety of related technologies, including stereo vision, laser scanners, and depth cameras, have been used to sense tree structure in the past. Nielsen et al. (2012) explored a calibrated-cameras system to reconstruct the peach tree for automated blossom thinning. Dong et al. (2018) measured the semantic parameters of trees using only an RGB-D imaging device.

They introduced a robust detection and fitting algorithm to estimate the trunk size, tree height, canopy volume, and fruit count. Based on the time-of-flight principle, a depth camera was used to obtain some depth images of tall spindle apple trees. This process served as a first step toward developing a method for identifying branches and developing robotic systems for pruning (Gongal, Amatya, Karkee, Zhang, Lewis, 2015). However, the reflective properties, color, and complexity of a scene exert considerable effects on the accuracy of a depth camera.

Considerable studies have demonstrated that laser scanning sensors are highly accurate and independent of environmental conditions. Integrating LiDAR sensor into sprayer systems has proven effective for achieving variable-rate application (Liu & Zhu, 2016). One study showed that the reconstruction and structural parameters of fruit trees can be easily obtained using LiDAR sensor with a developed point cloud data processing method (Chakraborty, Khot, Sankaran, Jacoby, 2019). Other studies have reported using a combination of visual and laser scanning sensors; among these studies are one centered on grapevine detection and canopy dense map generation (Grocholsky, Nuske, Aasted, Achar, Bates, 2011) and another focused on tree trunk detection for different trellis-structured fruit trees (Bargoti, Underwood, Nieto, Sukkarieh, 2015; Xue, Fan, Yan, 2018). The aforementioned studies have demonstrated successful segmentation in the identification of individual trees and trunks using LiDAR sensors; however, the trellis wires and poles were ignored or eliminated as noise in these studies.

The primary goal of this study was to estimate the major specifications of high-density apple tree rows with a 3D LiDAR sensor. To achieve this goal, three specific objectives were defined and met: (1) a 3D LiDAR based imaging system was developed to acquire tree canopy point cloud data; (2) a semantic segmentation-based algorithm was developed to extract and label the trellis wires, support poles, and tree trunks using the acquired sparse 3D point cloud data; and (3) canopy width and depth maps were generated for the tested trees.

2. Materials and Methods

2.1 Sensor and Field Site

To acquire point cloud data for the tree canopy structure, a sensing system was attached to a utility vehicle driven manually through an orchard (Figure 1). The vehicle was driven about 0.5 m s^{-1} at the centre of two tree rows. The sensing system included a VLP-16 LiDAR sensor (Velodyne LiDAR, San Jose, CA, USA), an interface box for data transmission and power conversion, and a computer. The LiDAR sensor has a range accuracy of ±30 mm. It was fixed on a vertical aluminium T-slot stand and then attached to the vehicle with a distance of 3 m between it and the centre of the targeted tree row. The vehicle was moving at a slow speed, and the orchard ground is relatively flat; therefore, the orientation of the sensor was assumed to be consistent. The VLP-16 is a 360° revolving-head 3D LiDAR with 16 laser beams vertically separated along a range of 30° with 2° of angular resolution. It rapidly spins to scan the surrounding environment, capturing information including the distance to objects in the field of view, the 3D coordinates for each point, and the intensity of these objects with respect to the scanning centre. The rotation rate can be programmed anywhere from 5 Hz to 20 Hz. A rate of 5 Hz was used in this study. An open-source viewer software program (VeloView 3.5.0, Veloydyn Lidar Inc., San Jose, CA, USA) was used for real-time visualisation and live data stream recording. The data analysis was conducted using a Dell G3 laptop computer with an Intel i7 processor running at 2.2GHz and 16GB of RAM.

The experiment was conducted at a research orchard that is part of The Pennsylvania State University's Fruit Research and Extension Center, which is located in Biglerville, PA. The trees are planted in a tall spindle architecture with four tiers of trellis wires. The trellis wires are 2.54 mm diameter steel wires. The trees are ten-year old *GoldRush* trees with inter-row spacing of 4 m and plant spacing of 1.5 – 1.8 m. The height of the trees ranges from 3.3 to 3.7 m, and the width of the canopy is between 1.4 and 1.8 m. The diameter of the tree branches varies from 5 to 30 mm. Steel poles with a diameter of 50 mm have been placed between every three or four trees to support the trees.

Fig. 1. 3D LiDAR-based scanning system used for data capture in a tall spindle apple tree orchard

A 3D point cloud data processing algorithm was developed in the MATLAB environment (2019a, MathWorks, Inc., Natick, MA, USA). This algorithm facilitated trellis wire detection, the identification of tree trunks and support poles, and the estimation of canopy density and depth maps. These data were processed automatically frame by frame using the algorithm.

2.2 Data Acquisition and Processing

2.2.1 Data Preparation

The utility vehicle with the LiDAR sensor was driven through the tree rows, and the point cloud data were recorded for a total of 69 trees on November 8, 2018, right after a harvest with a full canopy. A total of 1,138 frames of point cloud data were captured with MATLAB. Individual image frames were subsequently imported into the MATLAB workspace. A coordinate system was defined to localise each point recorded by the sensor with the origin point at the centre of the LiDAR sensor; namely, the x-axis was along the tree row, the y-axis was perpendicular to the tree row, and the z-axis extended upward vertically. Normally there are about 4,500 to 8,300 points in an image frame, including the points for the orchard ground, and the region of interest (ROI) for a frame is [-2, 2], [2.3, 4.5], and [-2, 2] in the x, y, and z directions, respectively (in metres). The VLP-16 sensor had been installed vertical to the ground during the test. Accordingly, in order to show the tree canopy in a normal orientation in the MATLAB environment, it was necessary to transform each coordinate of point cloud data using a matrix that turned the x-axis, y-axis, and z-axis of the recorded cloud data point into a y-axis, z-axis, and x-axis, respectively, in the defined coordinate system.

The points within an image frame do not only include the targeted tree row; they also include the ground and trees from other rows. To remove the untargeted points, an ROI containing the targeted tree row was selected. Using this method, trees and objects from other rows could be easily removed. The next object that was removed was the ground. The RANdom SAmple Consensus (RANSAC) is an iterative method of estimating the parameters of a mathematical model from a set of observed data that contains outliers. In an image, points corresponding to the ground have clear outliers with certain consistency; therefore, the ground is a good example of how to use the RANSAC algorithm. To remove the ground plane, the MATLAB function *pcfitplane*() was used. This function uses the M-estimator SAmple Consensus (MSAC) algorithm to find a plane. The MSAC algorithm is a variant of the RANSAC algorithm (Torr & Zisserman, 2000). Using the fitted plane, the ground points were segmented and removed from the 3D point cloud image.

2.2.2 Trellis Plane and Trellis Wire Detection

For a high-density tree orchard, a trellis system is installed alongside the tree rows to support the trees. Normally it forms a plane with support poles and wires at each row. The points of the support poles in a point cloud image are most likely in a line with high density due to their large diameter. Since the angle between two laser beams is 2°, two horizontal adjacent points in an image frame are ~0.1 m apart when the distance of the sensor to the object is 3 m. Meanwhile, the distance between two vertical adjacent points is about 14 mm when the sensor is 3 m away from the object, since the angular resolution in the vertical direction is 0.4°. Considering the position of the trellis wire (in the horizontal direction, typically with no vertically adjacent points), a radius of 0.1 m was selected to determine an isolated point. If a point had fewer than 4 neighbouring points, it was categorised as an isolated point.

Since the trellis wires are all in a plane and parallel to each other, the isolated points representing the trellis wires should be within this same plane. Similar to the ground plane detection, considering the range accuracy, all the isolated points within a distance of 0.1 m were included to generate the trellis plane in the *y*-axis direction. The trellis plane is a vertical plane along the tree rows that is parallel to the *x*-*z* plane. The isolated points outside of the trellis plane were identified as a part of the tree canopy. If there are not enough isolated points to fit a plane or the fitted plane is not in an appropriate position, no trellis plane is established in a given image frame. Under these circumstances, we can use the trellis plane formed from adjacent frames. Trellis wires are parallel to each other along the tree rows, and the points for these trellis wires should have similar coordinate values at the *y*-axis. The *y*-axis value for each of the isolated points within the trellis plane was set to the points' average value. Finally, the MSAC algorithm was used to fit the trellis wire lines within the identified trellis plane.

2.2.3 Tree Trunk and Support Pole Extraction

The LiDAR sensor was installed vertically to cover the full tree canopy when driven along the tree row, thereby reducing the point cloud density in the horizontal direction (along the tree row). Tree trunks and trellis support poles have a similar point cloud in a frame due to their similar straight-up orientation. The diameter of the tree trunks is about 80 –100 mm and that of the poles is 50 mm. Although the VLP-16 LiDAR sensor is able to sense 16 beams at a time, it is possible that only one or none hit on a tree trunk or a pole in a single scene frame. With one laser beam targeting, the tree trunk or pole appears as a dotted line. No tree trunk or pole appears in the frame if the laser beam does not hit one of them during this interval.

In contrast to branches and leaves with their irregular shapes, both tree trunks and support poles have a cylinder-like shape. Given this geometrical consistency and the point cloud data measured by the VLP-16 sensor, the tree trunks and support poles were able to be extracted as follows.

A 3D point cloud frame was divided into 16 2D point sets according to the 16 different beams (laser ID). For a given laser beam, only the *y-z* coordinating plane was considered. In addition, tree trunks or poles are typically located close to the trellis plane; thus, the range of *y* coordinate values for them can be defined as [$Ypmin$-δ, $Ypmax$+δ], where *Ypmin* and *Ypmax* are the minimum and maximum *y*-axis values of the trellis plane. As the diameter of each tree trunk was about 100 mm and each tree trunk was growing slightly off-centre, it was necessary to set the margin constant δ to 0.2 m to guarantee the inclusion of all tree trunks and poles.

Next, suspension areas with high-density clusters of points with noise were identified. For a set of points $X(y, z)$ in a given beam, the function *dbscan*(), or a density-based spatial clustering of applications with noise, was applied to identify the high-density point areas, as shown in Equation 1,

$$labels = dbscan\ (X, epsilon, minpts) \quad (1)$$

where *labels* is the cluster index of the high-density point areas, *X* is the input point sets, *epsilon* is a numeric scalar that defines a neighbourhood search radius around a point, and *minpts* is the minimum number of neighbours required for a core point. In this study, the *epsilon* was set to 0.1 m, and 30 was selected for the *minpts*.

An analysis of these high-density candidate clusters was then performed to determine the tree trunks and support poles. Given the features of tree trunks and support poles, the point clusters of these objects should be presented as continuous, vertical, and highly dense in a thin volume. Tree trunks are located near the ground, while support poles may extend to much higher locations. The LiDAR sensor was 1.5 m above the ground, and the bare portion of each tree trunk was normally less than 1 m. To include all the tree trunks, a z-axis value of -0.5 m was set for the upper margin of the tree trunks. The boundary of these candidate clusters was plotted using the *boundary*() function in MATLAB. The area of each cluster within the boundary and the height-width ratio (HWR) of the candidate clusters were calculated. The density of the points for these clusters was defined as the number of points divided by the area. After attempting different values for the point cloud density and HWR, some constraint values were used to determine the tree trunks and support poles. In this study, tree trunks were identified with the constraints of z-axis values lower than -0.5 m, a point cloud density greater than 3,500, and an HWR greater than 1.5. Meanwhile, a support pole was identified with a point density greater than 4,000, an HWR greater than 3, and no constraint for the z-axis value. Tree trunks and poles cannot be scanned by the same beam in an image frame. Therefore, in a given beam, a long vertical line segment can be labelled a support pole. A shorter vertical segment with a smaller y-axis value can be labelled a tree trunk. If a point cluster did not qualify as either of these, the point cluster was categorised as tree canopy.

The final step was to mark and record these tree trunks or pole points and repeat the previous three steps for the other 15 beams of point cloud data.

2.3 Method of Validating Results

In order to assess the performance of the segmentation for all three types of objects, precision, recall, and accuracy were used, as presented by Olson & Delen (2008):

$$Precision = \frac{T_P}{T_P + F_P} \quad (2)$$

$$Recall = \frac{T_P}{T_P + F_N} \quad (3)$$

$$Accuracy = \frac{T_P + T_N}{T_P + T_N + F_P + F_N} \quad (4)$$

In Eqs. (2) to (4), T_P, T_N, F_P, and F_N are the number of true positive, true negative, false positive, and false negative counts of each image frame. We used Eqs (2) to (4) to calculate the precision, recall, and accuracy of the trellis wire segmentation, as well as the segmentation of the support poles and tree trunks. Take the trellis wires as an example: A true positive indicates that the segmented object is a real trellis wire, a true negative means no trellis wire object was identified in the image frame by either the algorithm or the manual method, a false positive means a non-trellis wire object was identified as a trellis wire, and a false negative suggests the real trellis wire was not identified as a trellis wire. The ground truth for each of the objects was extracted manually.

2.4 Canopy Density and Depth Maps

After the segmentation of the trellis wires, support poles, and tree trunks is complete, the remaining points within a point cloud image can be categorised as the tree canopy. By processing these remaining points, some key canopy specifications, such as canopy height, density, and depth, can be obtained.

In contrast to a 2D LiDAR sensor, the 3D VLP-16 LiDAR produces a sparse point cloud. The point densities of the poles and trunks are much larger than that of tree canopy. To provide a precise canopy structure, the points of the tree trunks and poles should be removed. Here is the procedure to identify the canopy density and depth maps.

The tree canopy can be divided into two sides with the trellis plane in the middle. The canopy on the LiDAR sensor side can be fully hit by the laser beams, while the other side can be hit only partially. Therefore, the LiDAR sensor is unable to represent the full canopy of trees accurately since it is scanning only one side of the trees. Only the half canopy on the LiDAR sensor side was considered and analysed in this study. Since the trellis plane was previously identified and was able to be used as a central plane for the tree canopy, it was fairly easy to extract the targeted half canopy between the boundary points and the trellis plane.

For tree height, only the z-axis value of each of the canopy points needs to be considered. The maximum and minimum z-axis values of the targeted points were obtained using the Max(z) and Min(z) functions in MATLAB. The height of the tree canopy was then calculated using Eq (5),

$$H = \mathrm{Max}(z) - \mathrm{Min}(z) \tag{5}$$

where Max(z) and Min(z) represent the maximum and minimum z-axis values of the canopy points.

In order to evaluate the overall canopy density and depth map, a point cloud image was gridded into equivalent-sized small areas. In this study, we set the size of each of these small areas as 10 or 20 cm squared. By counting the number of points in these areas, we were able to calculate the canopy density, as shown in Eq. (6),

$$Density = \frac{N}{S} \quad (6)$$

where N is the number of points in a fixed region and S is the area of the fixed region in m^2.

The depth map of a tree canopy integrates depth information from each of these small defined areas. The depth of each area can be calculated as the *y*-axis value of the trellis plane minus the minimum *y*-axis value of the points in the given small area, as shown in Eq. (7),

$$Depth = y_{trellis\ plane} - \mathrm{Min}(y) \quad (7)$$

where $y_{trellis\ plane}$ is the *y*-axis value of trellis plane and $\mathrm{Min}(y)$ is the minimum *y*-axis value of the points in the defined area. To consider both the maximum canopy depth and the density of a certain area, the average depth of each area was calculated by taking the *y*-axis value of the trellis plane and subtracting the average *y*-axis value of all the points in the small area, as shown in Eq. (8),

$$Depth_{av} = y_{trellis\ plane} - \frac{\sum_{i=1}^{n} y_i}{N} \quad (8)$$

where $y_{trellis\ plane}$ is the *y*-axis value of the trellis plane, y_i is the *y*-axis value of the i^{th} point in the targeted area, and N is the total number of points in the targeted area.

3. Results and Discussion

3.1 Trellis Plane and Trellis Wire Segmentation

A total of 1,138 frames of 3D point cloud images were processed to identify the trellis wires and trellis plane. An example of an image frame and the processing procedure is given in Fig. (2). As shown in Fig. 2(a), the raw 3D point cloud acquired by the 3D LiDAR was presented in a 3D coordinate system. The point cloud in this figure may include the points for the orchard ground (at the bottom), as well as the canopy points (CP), support pole points (SPP), trellis wire points (TWP), and tree trunk points (TTP). With the help of the MSAC algorithm, the ground plane was detected. The ground points were segmented and are presented in yellow in Fig. 2(b).

In the test, the acquired point cloud was sparse due to the vertical installation of the VLP-16 LiDAR sensor. Per the definition of isolated point (in section 2.2.2), the isolated points in the frame were identified (the red star dots in Fig. 2(c)). The specification of the trellis wires means that they are more regularly shaped than the other objects in the frame. The isolated points for the trellis wires can be easily identified. A horizontal trellis wire may be some discrete points in a line. Using the MSAC algorithm, therefore, a trellis plane can be fitted to include these isolated points of the trellis wires (shown in Fig. 2(d)). The position of the trellis plane is very useful information for pole or trunk detection, as well as density or depth map generation. However, in some of the image frames, the trellis plane could not be fitted due to the identification of very few isolated points. In these frames, the majority of the trellis wires may have been blocked by tree branches and leaves. Given that five frames were recorded every second and the speed of driving was relatively slow, a single tree may have been included in multiple frames. There is no need to obtain the trellis plane for each of the frames. A trellis plane can still be created for a particular tree canopy using one of these neighbouring frames.

In Fig. 2(d), the trellis wires are presented clearly with two dotted lines. Using the MSAC algorithm, the dots for the trellis wires were extracted and the other dots were eliminated, as shown in Fig. 2(e). Trellis wires are the lines in the horizontal direction that are almost parallel to the ground surface. Other lines with different orientations could have also been extracted, but they are not trellis wires and were ignored. In addition, as the trellis wires are spaced at fixed intervals due to the design of the trellis system, some suspension lines in the wrong position were also excluded from consideration. Based on the aforementioned selection criteria, all the trellis wires were extracted with fitted lines, as shown in Fig. 2(f).

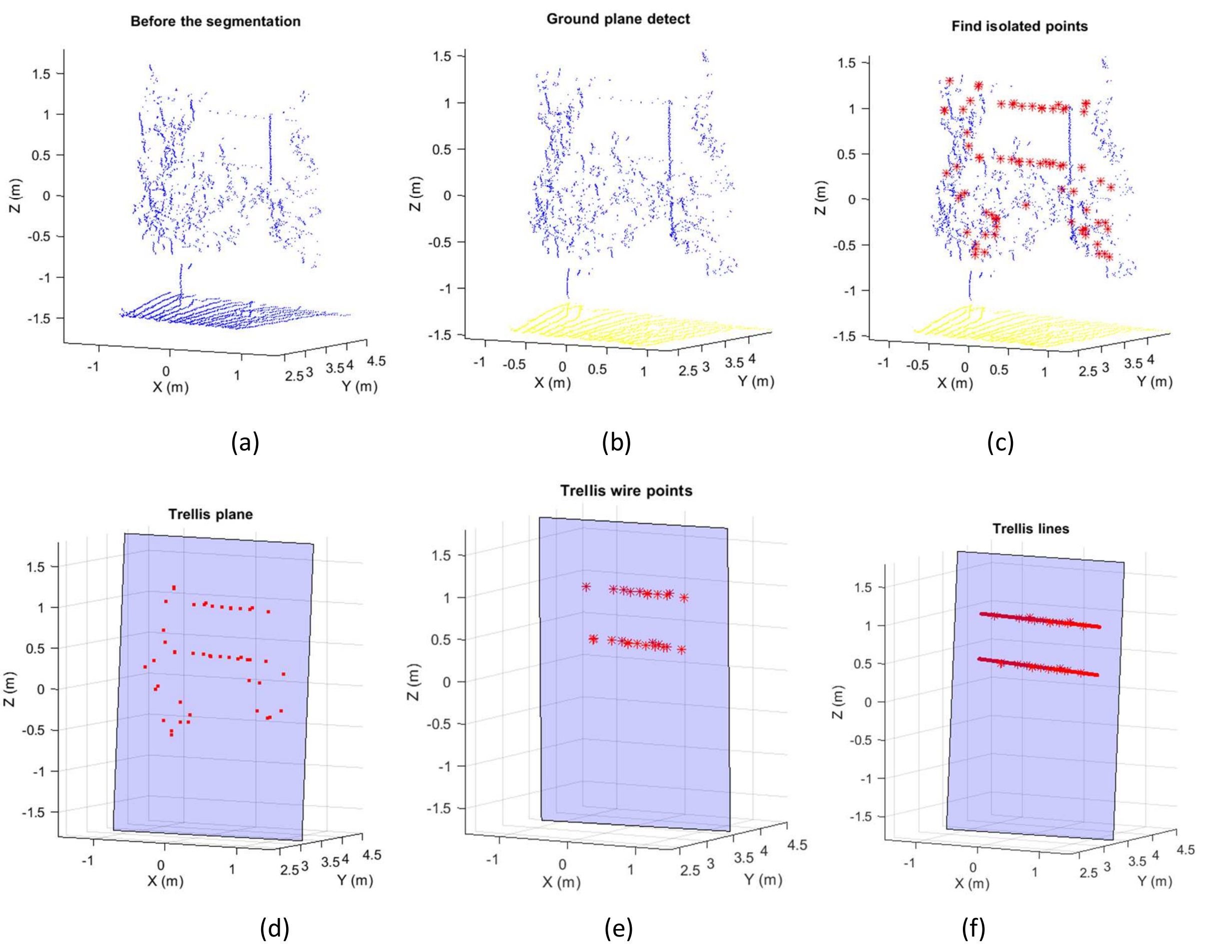
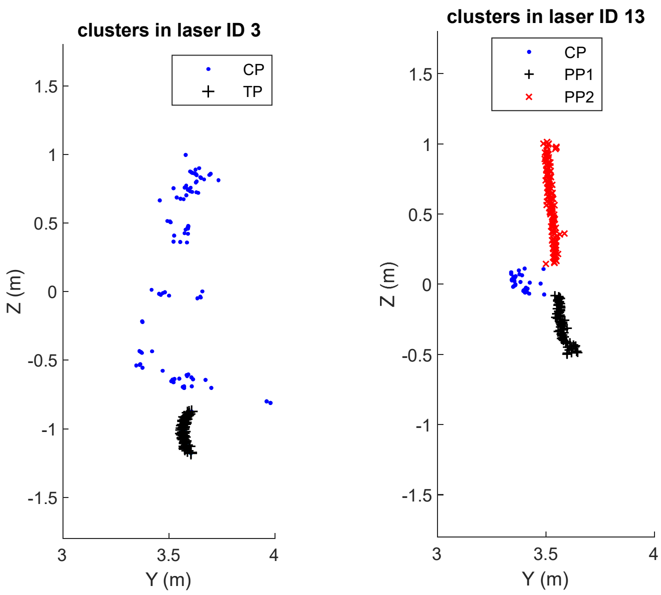


Fig. 2. Trellis plane and trellis wire extraction: (a) 3D point cloud before segmentation, (b) ground plane detection (yellow points are ground points), (c) isolated points in the point cloud, (d) isolated points in the trellis plane, (e) TWP in the trellis plane, and (f) trellis wires and trellis plane.

3.2 Tree Trunk and Support Pole Segmentation

Unlike trellis wires, tree trunks and support poles appear vertically upward in relation to the ground surface. Using the tree trunk and support pole extraction method described earlier, the most high-density line segments in a given beam (as distinguished by laser number) can be obtained. Fig. 3(a) is a point cloud image scanned by laser ID 3 showing a *y-z* plane containing a tree trunk marked with a black + and canopy points marked with blue •.

Figure 3(b) shows two high-density lines segmented in the same image frame as Fig. 3(a) but scanned by laser ID 13. According the judging criteria described earlier, the two lines belong to a support pole. The support pole was separated into two line segments marked with a red × and a black +, respectively, which can be mainly attributed to the occlusion of the branches and leaves. Based on the figure, the bottom portion of the pole is missing for the same reason. Nevertheless, the information is sufficient to identify the lines as a support pole.



(a) (b)

Fig. 3. Tree trunk and support pole segment segmentation according to different laser IDs: (a) tree trunk points (TP, marked with black +) segmentation and (b) support pole points (PP, marked with red × and black +) segmentation. Canopy points are marked with blue ·.

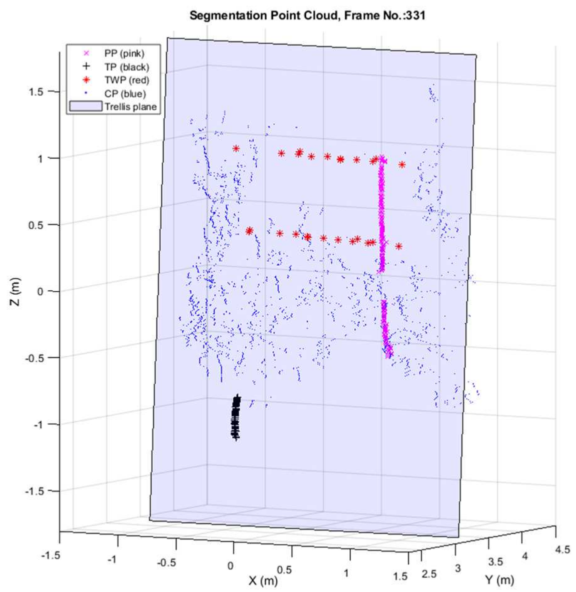


Fig. 4. Semantic segmentation results of a tree canopy: PP (pink ×), TP (black +), TWP (red*), and CP (blue·) segmentation and trellis plane (light blue rectangle).

Using the aforementioned steps, the trellis wires, support poles, and tree trunks were all segmented from the point cloud images. The remaining points that were neither trellis wires (Fig. 2) nor tree trunks or support poles (Fig. 3) were categorised as part of the tree canopy. Having completed the integration of these segmented objects, an overview of a section of the targeted tree row is illustrated in Fig. 4.

3.3 Validation of Results

In total, 1,138 frames of point cloud images were processed with the developed algorithm to segment the targeted objects. Meanwhile, each frame was labelled manually for the ground truth. The performance of the developed algorithm in identifying trellis wires, support poles, and tree trunks is shown in Table 1. The segmentation accuracy values for identifying the three types of objects were 88.6%, 82.1%, and 94.7% respectively. The segmentation accuracy value for the support poles was the lowest. In terms of precision, the precision value for the support poles was only 68.5%, which resulted in low segmentation accuracy. Among these segmented support poles are some thick, long, and vertical tree branches, which were inadvertently included because of similar characteristics such as high-density points. These incorrectly categorised frames increased the number of false negatives, ultimately resulting in low accuracy. Among the three types of point segmentation, tree trunks yielded the highest accuracy. In the study, only the bottom portion of the tree trunk was targeted. Since there are very few branches at the bottom of a tree, the developed algorithm was able to achieve very high accuracy in identifying the tree trunks. Ideally, the tree trunk (bottom portion) and the support poles would be standing straight upward. When tilted, these objects may not be hit by the laser beam in the same image frame, resulting in the incorrect segmentation of these objects. The situation could be improved if larger-sized support poles were used. In our method, the trellis plane was extracted first; the other objects were then segmented based on their relative location to the plane. In the future, we plan to combine LiDAR sensor with a depth camera to investigate the possibility of improving tree orchard object detection accuracy.

Table 1. Results of segmentation for three different types of objects in tree rows

	Precision	Recall	Accuracy
Trellis wires	95.2%	89.7%	88.6%
Support pole	68.5%	98.1%	82.1%
Tree trunk	96.3%	97.5%	94.7%

The leaf area estimation of fruit trees using machine vision technologies has been widely investigated (Sanz-Cortiella et al., 2011; Mora et al., 2016). Meanwhile, there are a few studies on tree trunk detection. For example, Bargoti et al. (2015) developed a pipeline for trunk detection in trellis-structured apple orchards. A satisfactory trunk detection accuracy of 87 – 96% was achieved during the pre-harvest season. The remaining 1 – 2 % error was due to support poles' inadvertent classification as tree trunks. The detection of the support poles and trellis wires could help identify the tree canopy for orchard operation tasks. To date, there have been no studies investigating the detection of support poles and trellis wires in the tree orchard, which was one of the major focuses of our study.

3.4 Canopy Density and Depth Maps

Using Eq. (6), canopy density was calculated for the tested tree canopies. Figure 5(a) shows the overall canopy density (with trellis wires and support poles) using a grid size of 0.1 × 0.1 m. After eliminating the grids for the trellis wires and support poles, a corrected canopy density map was obtained, as shown in Fig. 5(b). We call this the leaf area density map. This map can provide information about the branch/leaf distribution in an overall tree canopy, thereby assisting with precision application in some production tasks. For example, more pesticides could be applied to high-density areas using a precision spraying system. The size of the grid could be adjusted based on application needs. For example, it could be set as 0.2 × 0.2 m. The corresponding leaf area density map is shown in Fig. 5(c).

Canopy depth is another important parameter for representing the characteristics of the tree canopy. Using Eq. (7), a canopy depth map with a grid size of 0.2 × 0.2 m was created, as shown in Fig. 5(d). The canopy depth map gives the overall outer boundary of a tree canopy, which could be useful in orchard tasks such as mechanical pruning. Given this information about the tree boundary, the position of the cutter on a pruning machine could be adjusted for precision and automated branch cutting. The average depth map is also provided in Fig. 5(e). The average depth map integrates the depth and density of the point cloud data. If there are more points at the inner canopy (close to the tree trunks), the value shown in the average depth map is smaller. If there are more points at the outer canopy, the value shown in the average depth map is larger.

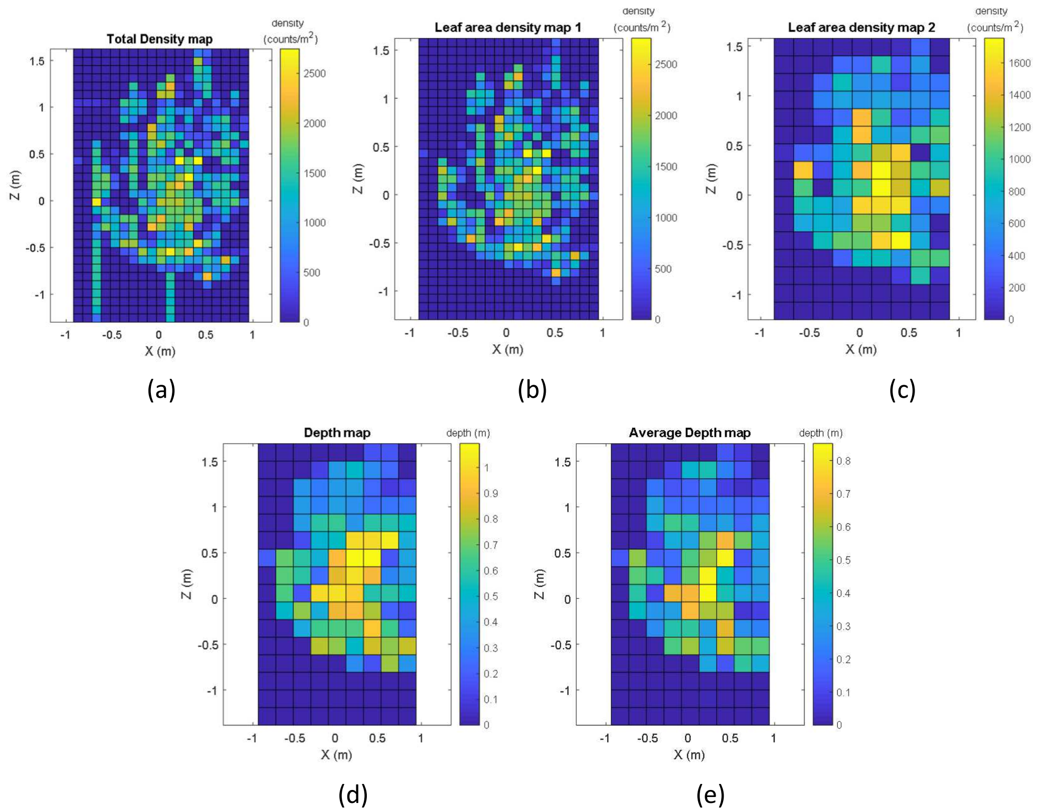


Fig. 5. Density maps and depth maps of a targeted tree canopy: (a) total point area ratio density map with PP, TWP, TP, and CP all considered; (b) point area ratio density map only considering CP with a grid size of 0.1×0.1 m; (c) point area ratio density map only considering CP with a grid size of 0.2×0.2 m; (d) depth map; and (e) average depth map with a grid size of 0.2×0.2 m.

In summary, with data collected using a 3D LiDAR sensor, the developed algorithm was able to segment the trellis wires, support poles, and tree trunks from the point cloud images with high accuracy. Furthermore, the canopy density map and depth map could also be created. While our study focused on trees in a high-density, trellis-trained system, the algorithm developed in this study may be applicable to other types of tree orchards with different row spacings, tree dimensions, and pole sizes. In orchards with much higher canopy density, however, it may not be easy to detect the trellis wires. The outcomes of the study could be used as guiding information for various orchard operation tasks. Tree trunks and poles could be used to identify the trellis plane. The fitted trellis plane with tree trunks and support poles could be used as a reference plane for orchard platform navigation. Finally, the trellis plane and canopy density and depth maps could provide critical information for variable spraying systems and automated mechanical pruning.

The developed algorithm mainly focused on identifying objects in the tree rows using 3D point cloud data processing. The algorithm is not limited to the VLP-16 LiDAR; it could be used for other types of LiDAR sensors as well. The 3D LiDAR we used in our study has only 16 channels (laser beams), and an image frame of the point cloud was made of the points of the tree canopy hit by these laser beams at the same duration. The LiDAR was installed to cover a full tree (about 2 m in width), which means there were only 16 beams at vertical orientation available to represent a tree in a single image frame. In our current algorithm, one image frame was processed at a time, which may have resulted in less plentiful points in the image frame for accurately identifying all the objects. It is possible to use 3D LiDAR sensor to obtain five image frames per second and multiple image frames for a single tree if the driving speed is low. To improve the accuracy of object segmentation, though, more points need to be obtained. Therefore, it is necessary to improve the algorithm so that it can process the integration of multiple neighbouring image frames. Meanwhile, an inertial navigation system, including an inertial measurement unit (IMU) and a real-time kinematic (RTK) global position system (GPS), should be added to the current system to improve its object detection accuracy in tree orchards. The IMU could be used to correct the orientation of the LiDAR sensor, especially in the case of an orchard situated on uneven terrain. The GPS could be used to record the relative location of each image frame, which would be useful for generating an accurate map of the entire orchard block. Meanwhile, deep learning methods could be tested for improving the accuracy of object segmentation in the orchard environment.

4. Conclusions

The purpose of this study was to estimate the shape/structure of apple tree canopies using a 3D LiDAR sensor system. An algorithm was developed to identify the major objects in a tree canopy system. Tree canopy density and depth maps were then created. The following conclusions can be drawn from this study:

1. The developed algorithm segmented the trellis wires, support poles, and tree trunks for the tested tree canopies with overall accuracies of 88.6%, 82.1%, and 94.7%, respectively.
2. The trellis plane could be computed from the point cloud, and used as the middle plane in the overall tree canopy.

3. After eliminating the trellis wires and support poles from the point cloud, the tree density and depth maps could be created.

This study provided a foundation for automatically detecting specific canopy objects in the apple tree rows of high-density orchards. The findings of this study indicate the proposed method's potential for improving automatic orchard operations.

Acknowledgments

This research was supported in part by the United States Department of Agriculture's (USDA) National Institute of Food and Agriculture Federal Appropriations (Project PEN04547; Accession No. 1001036), the State Horticultural Association of Pennsylvania (SHAP), the Natural Science Foundation of Hebei Agricultural University (Grant No. ZD201701), and a study-abroad program for young teachers sponsored by Hebei Agricultural University.

References

Bargoti, S., Underwood, J. P., Nieto, J. I., & Sukkarieh, S. (2015). A Pipeline for Trunk Detection in Trellis Structured Apple Orchards. *Journal of Field Robotics, 32*(8), 1075-1094.

Chakraborty, M., Khot, L. R., Sankaran, S., & Jacoby, P. W. (2019). Evaluation of mobile 3D light detection and ranging based canopy mapping system for tree fruit crops. *Computers and Electronics in Agriculture, 158*, 284-293.

Colaço, A. F., Molin, J. P., Rosell-Polo, J. R., & Escolà, A. (2018). Application of light detection and ranging and ultrasonic sensors to high-throughput phenotyping and precision horticulture: current status and challenges. *Horticulture Research, 5*(1), 35.

Dong, W., Roy, P., & Isler, V. (2018). Semantic Mapping for Orchard Environments by Merging Two-Sides Reconstructions of Tree Rows. *arXiv:1809.00075*.

Fernández-Sarríaa, A., López-Cortés, I., Estornell, J., Velázquez-Martí, B., & Salazar, D. (2019). Estimating residual biomass of olive tree crops using terrestrial laser scanning. *International Journal of Applied Earth Observation and Geoinformation, 75*, 163-170.

Gongal, A., Amatya, S., Karkee, M., Zhang, Q., & Lewis, K. (2015). Sensors and systems for fruit detection and localization: A review. *Computers and Electronics in Agriculture, 116*, 8-19.

Grocholsky, B., Nuske, S., Aasted, M., Achar, S., & Bates, T. (2011). A camera and laser system for automatic vine balance assessment. Paper presented at the American Society of Agricultural and Biological Engineers, Louisville, Kentucky.

He, L., & Schupp, J. (2018). Sensing and Automation in Pruning of Apple Trees: A Review. *Agronomy, 8*(10), 211.

Li, L., He, X., Song, J., Liu, Y., Zeng, A., & Yang, L. (2018). Design and experiment of variable rate orchard sprayer based on laser scanning sensor. *International Journal of Agricultural and Biological Engineering, 11*(1), 101-108.

Liu, H., & Zhu, H. (2016). Evaluation of a Laser Scanning Sensor in Detection of Complex-Shaped Targets for Variable-Rate Sprayer Development. *Transactions of the ASABE, 59*(5), 1181-1192.

Lordan, J., Francescatto, P., Dominguez, L. I., & Robinson, T. L. (2018). Long-term effects of tree density and tree shape on apple orchard performance, a 20 year study—Part 1, agronomic analysis. *Scientia Horticulturae, 238*, 303-317.

Mora, M., Avila, F., Carrasco-Benavides, M., Maldonado, G., Olguín-Cáceres, J., & Fuentes, S. 2016. Automated computation of leaf area index from fruit trees using improved image processing algorithms applied to canopy cover digital photograpies. *Computers and Electronics in Agriculture*, *123*, 195-202.

Nielsen, M., Slaughter, D. C., & Gliever, C. (2012). Vision-Based 3D Peach Tree Reconstruction for Automated Blossom Thinning. *IEEE Transactions on Industrial Informatics, 8*(1), 188-196.

Olson, D. L., & Delen, D. (2008). *Advanced Data Mining Techniques*: Springer Berlin Heidelberg.

Ontario Apple Grower. (2015). Best Management Practices for Building Trellis Support Systems for High Density Ontario Apples (Vol. 2019).

Palleja Cabre, T., Llorens, J., & Landers, A. J. (2017). Measuring crop canopy – the development of a dynamic system for precision fruit crop spraying. *Advances in Animal Biosciences, 8*(2), 250-254.

Sanz-Cortiella, R., Llorens-Calveras, J., Arnó-Satorra, J., Ribes-Dasi, M., Masip-Vilalta, J., Camp, F., Gràcia-Aguilá, F., Solanelles-Batlle, F., Planas-DeMartí, S., Pallejà-Cabré, T., & Palacin-Roca, J. 2011. Innovative LIDAR 3D dynamic measurement system to estimate fruit-tree leaf area. *Sensors*, *11*(6), 5769-5791.

Tagarakis, A. C., Koundouras, S., Fountas, S., & Gemtos, T. (2018). Evaluation of the use of

LIDAR laser scanner to map pruning wood in vineyards and its potential for management zones delineation. *Precision Agriculture, 19*(2), 334-347.

Torr, P. H. S., & Zisserman, A. (2000). MLESAC: A New Robust Estimator with Application to Estimating Image Geometry. *Computer Vision and Image Understanding, 78*(1), 138-156.

Wandkar, S. V., Bhatt, Y. C., Jain, H. K., Nalawade, S. M., & Pawar, S. G. (2018). Real-Time Variable Rate Spraying in Orchards and Vineyards: A Review. *Journal of The Institution of Engineers (India): Series A, 99*(2), 385-390.

Xue, J., Fan, B., & Yan, J. (2018). Trunk detection based on laser radar and vision data fusion. *International Journal of Agricultural and Biological Engineering, 11*(6), 20-26.