2950 Niles Road, St. Joseph, MI 49085-9659, USA
269.429.0300 fax 269.429.3852 hq@asabe.org www.asabe.org

An ASABE Meeting Presentation
DOI: https://doi.org/10.13031/aim.202100428
Paper Number: 2100428

Plant organ segmentation from point clouds using Point-Voxel CNN

Farah Saeed[1], Changying Li[2]
[1]Department of Computer Science, University of Georgia, Athens, GA 30602, USA
[2]College of Engineering, University of Georgia, Athens, GA 30602, USA

Written for presentation at the
2021 Annual International Meeting
ASABE Virtual and On Demand
July 12–16, 2021

ABSTRACT. *Plant phenotyping is a vital process in plant breeding studies and crop improvement. Automated segmentation of plant organs is a challenging task for conventional images due to occlusions. Three-dimensional (3D) point cloud data can overcome the challenge as the 3D data provides depth information and minimizes occlusion. The goal of this study is to perform plant organ segmentation from point clouds using 3D deep learning method called Point-Voxel Convolutional Neural Network (PVCNN). PVCNN utilizes point-based and voxel-based 3D data representation to achieve computational efficiency that results in higher speed up and lower memory consumption compared with other 3D deep learning approaches. We use cotton plants to predict main stem, branches, and cotton bolls. Point cloud data were collected using a LiDAR scanner. We manually annotated point clouds using Open3D library. After training, PVCNN achieved mIOU and accuracy of around 81% and 92.7%, respectively. After postprocessing, the segmented organs were used in extracting phenotypic traits. Comparison of ground truth and predicted segments demonstrated an R squared value of 0.9 for the main stem's height and diameter after excluding outliers. The number of nodes achieved an R squared value of 0.6 and a root mean square error of average internode distance was less than 0.1 cm. This plant organ segmentation method based on a 3D point-voxel CNN makes it possible to measure phenotypical traits related to plant architecture, which is important for plant breeding and physiology.*

Keywords. *Deep learning, LiDAR, Plant organ segmentation, Point cloud, PVCNN.*

Introduction

Cotton is the most widely used non-food crop in the world (Voora, Larrea et al. 2020). Providing versatility, performance, and comfort, it is used in textiles, animal feed, cosmetics, pharmaceuticals, home furnishings, and other industrial applications. Plant phenotyping plays an important role in enhancing crop yield. Plant breeders perform plant phenotyping to study the morphological and physiological traits of crops for selecting suitable cultivars. Common phenotypic traits of interest are main stem's diameter, plant volume, branch angle, and length. While the manual measurement of phenotypic traits is time consuming and labor intensive, recent advancements in remote sensing technologies are gaining attention among plant researchers for efficiently visualizing plant architecture and measuring phenotypic traits.

Using two-dimensional (2D) images, plant scientists have achieved high-throughput phenotyping non-invasively. For

instance, researchers extracted phenotypic traits from images of Maize, Sorghum, wheat and soybean crops, and cranberry (Bai, Ge et al. 2016, Diaz-Garcia, Covarrubias-Pazaran et al. 2018, Atefi, Ge et al. 2019). For two-dimensional data collection, a range of different sensors exists. These include RGB cameras, hyperspectral cameras, thermal cameras, near infrared cameras and others. Data captured using an hyperspectral camera was utilized to detect disease in tobacco, and potato plants, respectively (Zhu, Chu et al. 2017, Polder, Blok et al. 2019). To capture multi-model plant traits, (Jay, Rabatel et al. 2015) integrated a multispectral, thermal IR LiDAR scanner on phenotyping platforms.

2D images from RGB sensors are inexpensive and support image processing functionalities. However, the suffer from occlusion and lack of depth information. This can be resolved with the use of three-dimensional (3D) data that can capture depth. Depth information makes it easier to estimate plant morphology, growth, and biomass by generating accurate coordinates, distance estimates, and the precise orientation of objects. There are several ways for recovering 3D data from 2D image information, including structure from motion and time of flight measurement. Previously, a 3D model using Structure from Motion was computed with RGB images to estimate plant structural parameters such as height and leaf area (Jay, Rabatel et al. 2015). A group used Multiview imaging systems and employed the 'shape of silhouette' method to extract 3D point clouds (Shi, van de Zedde et al. 2019). The methods for extracting 3D information from 2D data have some limitations in terms of accuracy. The methods may incur data loss in 3D when combining 2D images. 3D data collected directly using scanners like a LiDAR provides a more accurate representation of real world. There are various sensors for 3D data collection, such as the LiDAR scanner, the Kinect sensor, and the Commuted Tomography CT scanner. Collected 3D data can be represented in various formats, including point clouds or grids of voxels. Point cloud-based representation is a permutation invariant. It stores 3D data as a list of points with their features. The voxel-based representation considers 3D data as grid of voxels and is dependent on the order of points. Compared with 2D data, 3D data has lower occlusion and avoids the lack of depth information. As a result, there has been much work involving phenotyping on 3D data. LiDAR data has been used for morphological trait extraction as well as plant response and growth analysis (Omasa, Hosoi et al. 2007, Herrero-Huerta, Lindenbergh et al. 2018, Sun, Li et al. 2018, Panjvani, Dinh et al. 2019). (Sun, Li et al. 2020) identified cotton bolls in Lidar data. X-ray data was utilized for phenotypic trait extraction of plants such as sorghum and grape vine plants (Mairhofer, Zappala et al. 2012, Tracy, Black et al. 2012, Metzner, Eggert et al. 2015, Li, Klein et al. 2019).

In terms of data processing, many computer vision methods for shape identification and scene interpretation in 2D images are extended to 3D data. For instance, cylinder fitting method and RANSAC was employed to identify the main stem in cotton plants (Sun, Li et al. 2020). Many machine learning methods have also been extended to 3D point clouds. The SVM approach was adopted for the phenotypic tasks of segmenting plant organ (Ziamtsov and Navlakha 2019, Dutagaci, Rasti et al. 2020, Sun, Li et al. 2020).

Deep learning is an important class of algorithms in computer vision for image interpretations. There has been rapid growth in the field of deep learning for object detection and segmentation in the last decade. VGG, Alexnet, Faster R-CNN, and Yolo are among the most efficient and accurate methods of computer vision in 2D data. Deep learning for 3D point clouds has been introduced and has flourished in the past few years. Pointnet is among the first architecture for segmentation and classification in 3D point clouds (Qi, Su et al. 2017). Pointnet++, DGCNN, PointCNN, ShellNet, RIConv were introduced in the class of point-based 3D deep learning methods (Qi, Yi et al. 2017, Li, Bu et al. 2018, Wang, Sun et al. 2019, Zhang, Hua et al. 2019, Zhang, Hua et al. 2019). Voxel-based deep learning methods for volumetric 3D deep learning (Wang, Liu et al. 2017, Le and Duan 2018, Zhou and Tuzel 2018, Wang and Lu 2019). Point-voxel CNN (PVCNN) combined the point-based and voxel-based representation for efficient performance (Liu, Tang et al. 2019).

Plant organ segmentation from 3D point clouds can prove useful for providing a near accurate estimate of plant and organ level phenotypic traits. With regards to plant organ segmentation on 3D data, photogrammetric mapping of cotton bolls using hybrid methods was performed on LiDAR data (Sun, Li et al. 2020). In another work, plant lamina was segmented using machine learning methods on point features (Ziamtsov and Navlakha 2019). Tensor-based classification and segmentation was performed showing its applicability to different species and plants (Elnashef, Filin et al. 2019). Plant organ segmentation of rose bushes was performed using X-ray and synthetic data (Dutagaci, Rasti et al. 2020).

In this study, we attempted to segment the plant organs of cotton plants from point clouds using a 3D deep learning network called Point-Voxel Convolutional Neural Network. Both point-based and voxel-based representation were utilized to perform the study. A 3D labelled dataset comprising 28 point clouds of cotton plants was organized. The performance of PVCNN on the segmentation of the main stem, branch, and cotton bolls is analyzed and compared with baselines of Pointnet and Pointnet++. On the achieved segmentation results, the phenotypic traits of plant organs are extracted and compared with the ground truth segments.

Materials and methods

Figure 1 describes overall workflow of cotton plant organ segmentation and trait extraction. The input point clouds were preprocessed and plant organ segmentation based on a 3D deep learning-based segmentation method were applied. The main stem, branches, and bolls of the cotton plant were segmented into red, green and blue. On the segmented organs, post processing was applied. Finally, phenotypic traits were extracted from the segmented plant organs.

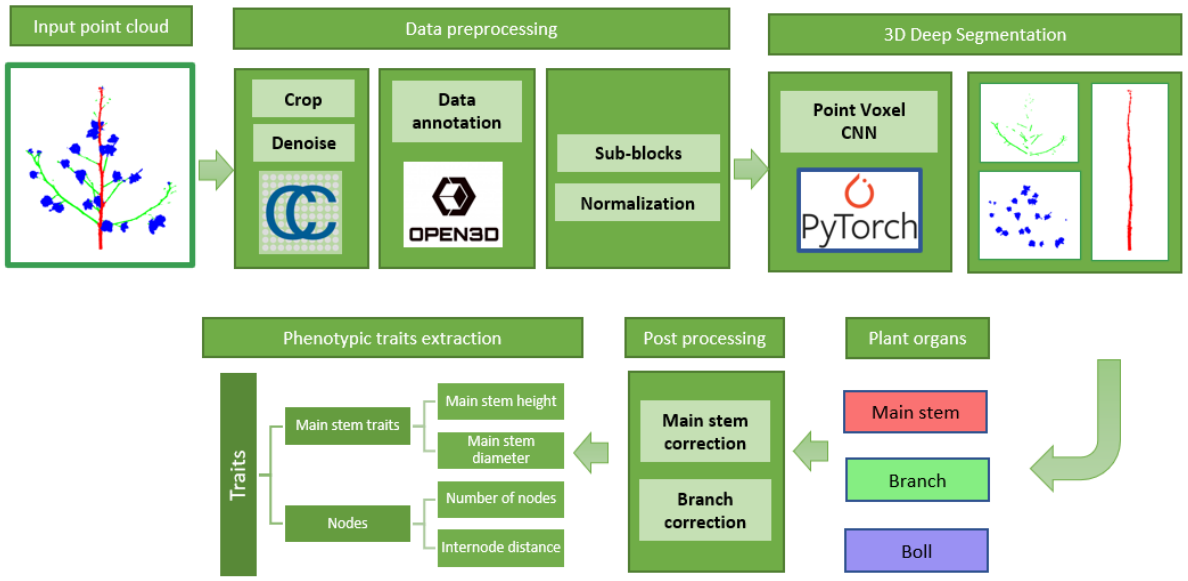

Figure 1: 3D Cotton plant organ segmentation and trait extraction workflow

Dataset

The three-dimensional (3D) point cloud data of cotton plants was collected in three sessions in the Plant Research Farm of the University of Georgia in USA. The cotton plants were post-defoliation. The first session was carried out in December 2018. In this session, point cloud data of cotton plants were collected from single plant plots. The second and third sessions were carried in December 2020 and February 2021, respectively. In these sessions, cotton plants in multi-plant plots were considered. The plants were cut from the bottom and taken to the barn. The plants were then stuck to a wooden platform to arrange them in a standing position. A Faro LiDAR scanner was used to collect the 3D data in all sessions. The LiDAR scans were registered to acquire the point clouds for the plants. A representation of the collected data is shown in Figure 2. The overall dataset comprises 28 point clouds of cotton plants. The detail for dataset from each session is summarized in Table 1. All point clouds contain *x*, *y*, and *z* coordinates and RGB information.

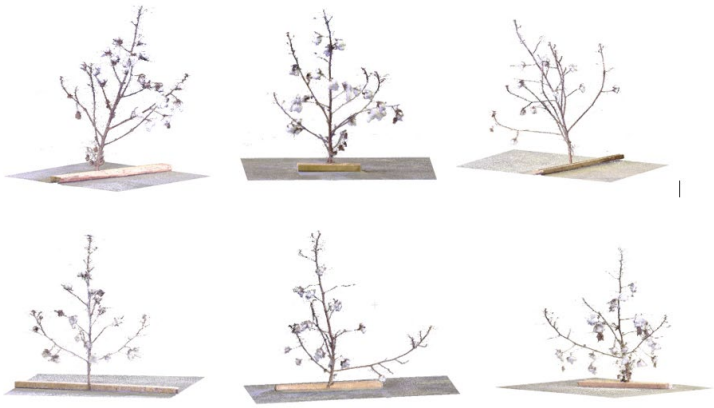

Figure 2: Examples of collected cotton plant point clouds

Table 1: Dataset summary

Data session	Session date	# of point clouds	Average # of points	Maximum # of points	Minimum # of points	Std dev of # of points	# of blocks
1	Dec 2020	10	621,984	953,922	396,055	171,619	612
2	Feb 2021	9	403,008	607,579	213,055	120,010	336
3	Dec 2018	9	406,189	480,466	295,635	52,586	807
Overall dataset		28	482,236	953,922	213,055	163,977	1,755

Preprocessing

The point cloud data were preprocessed before applying segmentation. Firstly, the ground plane points in the point clouds

were removed. This was achieved by using cropping function in the open source CloudCompare software. Figure 3(b) illustrates the resulting point cloud. The collected laser scans contain some noise which can lead to erroneous values for estimation of point cloud characteristics. To this end, the cropped point clouds were denoised using a statistical outlier removal method. Figure 3(c) illustrates the point cloud acquired after denoising.

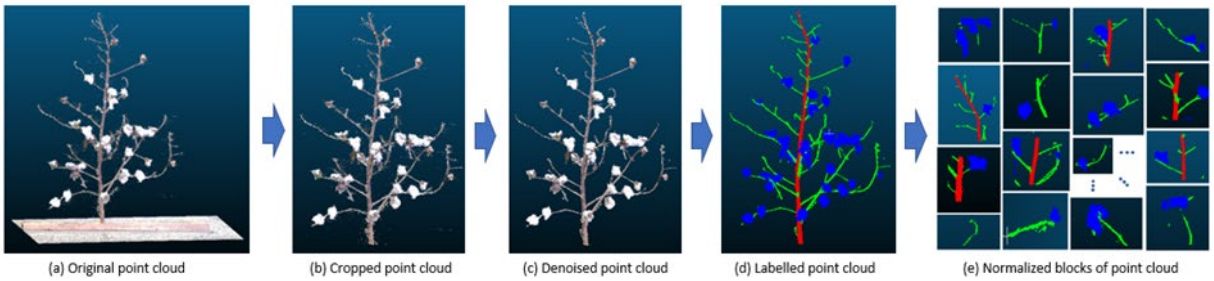

Figure 3: Preprocessing applied on input point cloud

For 3D deep segmentation of cotton plants, a labelled dataset was created by annotating plant organs in the point cloud. This was achieved using the Open3D library, an open-source library for 3D data processing that provides several functionalities, including cropping, visualizing, and the registering of point clouds. To allow the annotation of plant organs, we programmed a module in C++ using the Open3D library. Through this module, the polygon selection feature was used to select a certain plant organ for annotation. The entire dataset was annotated using this program. The main stem, branches, and cotton bolls in all point clouds were labelled in red, green, and blue, respectively. An example of an annotated point cloud is represented in Figure 3(d).

The 3D deep learning architectures for point clouds accept a fixed number of points in point clouds. The cotton plant point clouds in the dataset have more than 100,000 points. Because of hardware limitations, feeding the entire point cloud to the network required a large sub-sampling rate, resulting in significant loss of geometric information. To remedy this, we partitioned the point clouds into blocks of fixed size as shown in Figure 3(e). The points in each block are gathered as separate point clouds. These sub-point clouds were used to train the network. A point cloud being tested is also preprocessed in the same way. For example, all sub-point clouds of a cotton plants were combined to achieve a full segmentation of that plant. We divided the point cloud into blocks of 2048 points. Blocks with less than 10 points were removed, and if the block had more than 2048 points, we randomly sampled the desired number of points. For blocks with fewer than 2048 points, we duplicated the points to achieve the desired number. Table 1 shows the number of blocks in the dataset. Each sub-point cloud is normalized to a unit sphere. This means that each sub-point cloud is denormalized back to its original state after the prediction for trait extraction. During the training, we randomly jitter the position of each point by a Gaussian noise with zero mean and 0.01 standard deviation to improve the robustness of the network.

3D Deep learning methods

In this section, we first provide an overview of Pointnet and Pointnet++ and then an introduction of PVCNN and its architecture.

Pointnet

Pointnet is among pioneering works for performing deep learning on unordered point sets. It is a unified architecture for classification and segmentation that learns pointwise features independently with several shared multi-layered perceptrons (MLP) and aggregates a global feature using a max symmetric aggregation function.

The network takes n points as input. It then performs input and feature transformations. It uses shared MLP layers to extract features, which are aggregated using a max pooling layer. The obtained global feature is used to output classification scores for k classes. For achieving segmentation, the global feature is concatenated with each of the point's features. The resulting features are used to output per point scores. Pointnet extracts the global features of a point cloud but does not extract features that represent local geometric information of points.

Pointnet++

As Pointnet extracts pointwise features independently using a shared MLP (multi-layer perceptron), it can capture neither the local geometry in point clouds nor the interaction between points. For this purpose, Pointnet++ was introduced as an extension to Pointnet for capturing wider context and richer local features. Pointnet++ extracts point features by aggregating local neighborhood information.

Pointnet++ employs set abstraction module for capturing local patterns. The set abstraction module achieves the objective in three steps: sampling, grouping, and pooling. First, the points in the point cloud are sampled using farthest point sampling. Each sampled point forms a group containing points within a certain radius. Afterwards, features from each group using Pointnet are extracted and are aggregated by using a max pooling layer.

Pointnet++ architecture comprises multiple set abstraction modules to extract features hierarchically. In order to perform

well in point clouds with varying density and scales, the authors of Pointnet++ introduced multi-scale grouping in set abstraction module. In multi-scale grouping, the features of the points are extracted using the neighborhood information at different scales. The extracted features at different scales are then concatenated which generates the geometric information of the neighborhood of the point at different radius levels.

For achieving segmentation results, features for subsampled points are mapped onto original points using interpolation and the fully connected layers are applied on the resulting features to output per point scores.

PVCNN

Point-voxel convolutional neural networks are designed to achieve efficient 3D deep learning by utilizing both point-based and voxel-based representation of input data (Liu, Tang et al. 2019). The network utilizes a PVConv layer in the architecture. The PVConv layer consists of point-based and voxel-based branch used to extract features from sparse and dense data representations, respectively. The voxel-based branch is used for course-grained neighbor aggregation to extract local features from low-resolution voxel grids, and point-based branch performs fine-grained feature extraction for individual points without considering neighborhood information. The features extracted from both branches are fused together to represent coarse-grained and fine-grained information.

For achieving voxel-based feature aggregation, the input point cloud is voxelized to a voxel resolution *r*. On the obtained voxel grids, features are aggregated by applying the series of 3D volumetric convolutions. Batch normalization and nonlinear activation is applied after each 3D convolution. The aggregated features are devoxelized back to the point cloud domain. For this purpose, interpolation methods are used, like nearest-neighbor interpolation or trilinear interpolation. In the point-based branch, the features are extracted by applying the MLP layers in the entire point cloud. The aggregated features from voxel branch representing neighborhood information are added to the individual point features.

The original PVCNN architecture for part segmentation is formed by replacing MLP layers in Pointnet with the PVConv layers. We made modifications to the original PVCNN architecture to improve performance on cotton plants. In the PVCNN, the PVConv layer originally fused the voxel-based and point-based features by addition. We updated this layer to fuse the point-based and voxel-based features by concatenation rather than addition. We increased the number of PVConv and fully connected layers by 1. The exact architecture of PVCNN is shown in Figure 4. Finally, the input point coordinates inside the voxel were normalized with respect to the other points in that voxel. As a result, each voxel in the first PVConv layer was normalized independently. This ensured that the input point coordinates are normalized with respect to their voxels. The resulting coordinates were used to extract voxel-based features.

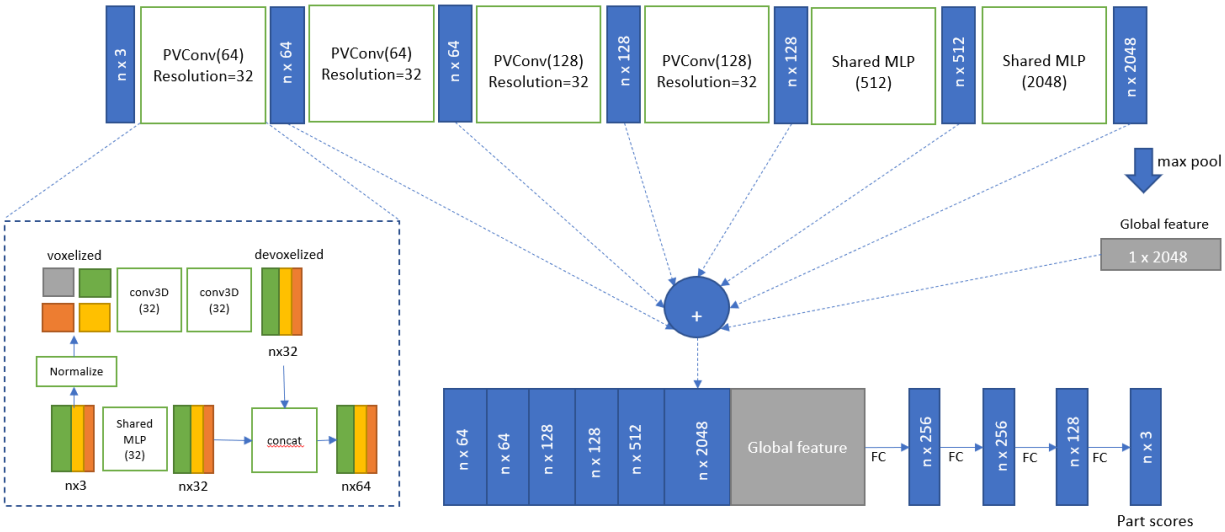

Figure 4: PVCNN Architecture. The network takes n points as input, applies feature extraction. It then aggregates the point features by max pooling. It concatenates the features to output scores per point. All layers are followed by batch norm and Relu. MLP is multi-layer perceptron on each point. FC is fully connected layer operating on each point. The PVConv layer applies 3D Convolution on voxelized input. It devoxelises the aggregated features back to map the points. The point-based and voxel-based features are concatenated to form the output.

Experiment Settings

For experiments, the dataset was split into train and test set with the ratio of 0.7 to 0.3. The number of point clouds, points, blocks, and distribution of classes in train and test set is given in Table 2. In training and testing the network, the point clouds were preprocessed into blocks. As shown in Table 2, the train set consisted of 1210 blocks and the test set consisted of 545 blocks. The blocks were fed to the network for training with a batch size of 32. Pointnet, Pointnet++, and PVCNN were trained for 500 epochs. All methods used an initial learning rate of 0.001. The Pointnet and Pointnet++ employed a step learning rate scheduler to reduce the learning rate by a factor of 0.5 every 20 epochs. In PVCNN, a cosine annealing learning rate scheduler was used. A Tesla V100 GPU card with 16GB memory was employed for all computation.

Table 2: Train and test dataset summary

Category	# of point clouds	Average # of point clouds	# of blocks	Distribution of classes (%)		
				Main stem	Branch	Boll
Train	19	485,808	1210	9	16	75
Test	9	474,696	545	13	19.5	67.5

Evaluation metrics

The performance of the PVCNN for plant organ segmentation of main stem, branch, and cotton bolls were evaluated using Precision, Recall, and Intersection over Union (IOU). The true positives, true negatives, false negatives, and false positives for each class are represented as TP_k, TN_k, FN_k, FP_k respectively where the equations *(1), (2)* and *(3)* are used for estimating Precision, Recall and IoU of cotton plant organs.

$$Precision = \frac{TP_k}{TP_k + FP_k} \quad (1)$$

$$Recall = \frac{TP_k}{TP_k + FN_k} \quad (2)$$

$$IOU = \frac{TP_k}{TP_k + FP_k + FN_k} \quad (3)$$

For estimating performance over all the classes, the mean IOU (mIOU) and total accuracy was used. The mIOU is calculated by averaging the IOU scores over all the classes. The accuracy is estimated as a ratio of all correctly classified points to the total number of points.

$$mIOU = \frac{1}{C}\sum_{k=1}^{C}\frac{TP_k}{TP_k + FP_k + FN_k} \quad (4)$$

$$Accuracy = \frac{\sum_{c=1}^{C} TP_k}{\sum_{c=1}^{C} TP_k + FP_k} \quad (5)$$

Results and discussion

Segmentation Results

The performance of the PVCNN on cotton plant organ segmentation was compared with 3D deep learning baselines of Pointnet and Pointnet++. As shown in Table 3, the PVCNN outperformed Pointnet and Pointnet++ in accuracy and mIOU. PVCNN has the highest performance because of its modified architecture. PVCNN also outperforms the baselines in Precision, Recall, and IoU, except in the case of a recall for branch class. The recall of the branch is higher in the case of Pointnet++. But Pointnet++ has a lower precision for branches as compared to PVCNN. The comparison of organ segmentation results of PVCNN with baselines and ground truth is given in (Figure 5). The IoU of the branch is highest in case of PVCNN. PVCNN achieves more than 85% IoU for the classes of main stem and bolls. In the case of Branch classes, the IoU, Precision, and Recall are quite lower compared to the other classes of main stem and bolls. This is because the shape of the branch is similar to the main stem.

Table 3: Performance evaluation of PVCNN with baseline

Metric	Classes	Pointnet	Pointnet++	PVCNN
Precision	Main stem	91.3	89.86	**93.48**
	Branch	64.11	77.06	**82.81**
	Bolls	81.68	93.61	**94.29**
Recall	Main stem	87.51	93.25	**94.03**
	Branch	64.68	**83.56**	79.8
	Bolls	83.04	92.57	**93.32**
IOU	Main stem	79.81	84.11	**87.81**
	Branch	42.81	66.51	**67.28**
	Bolls	69.18	86.8	**88.28**
mIOU		63.93	79.14	**81.12**
Accuracy		75.83	92.68	**92.77**

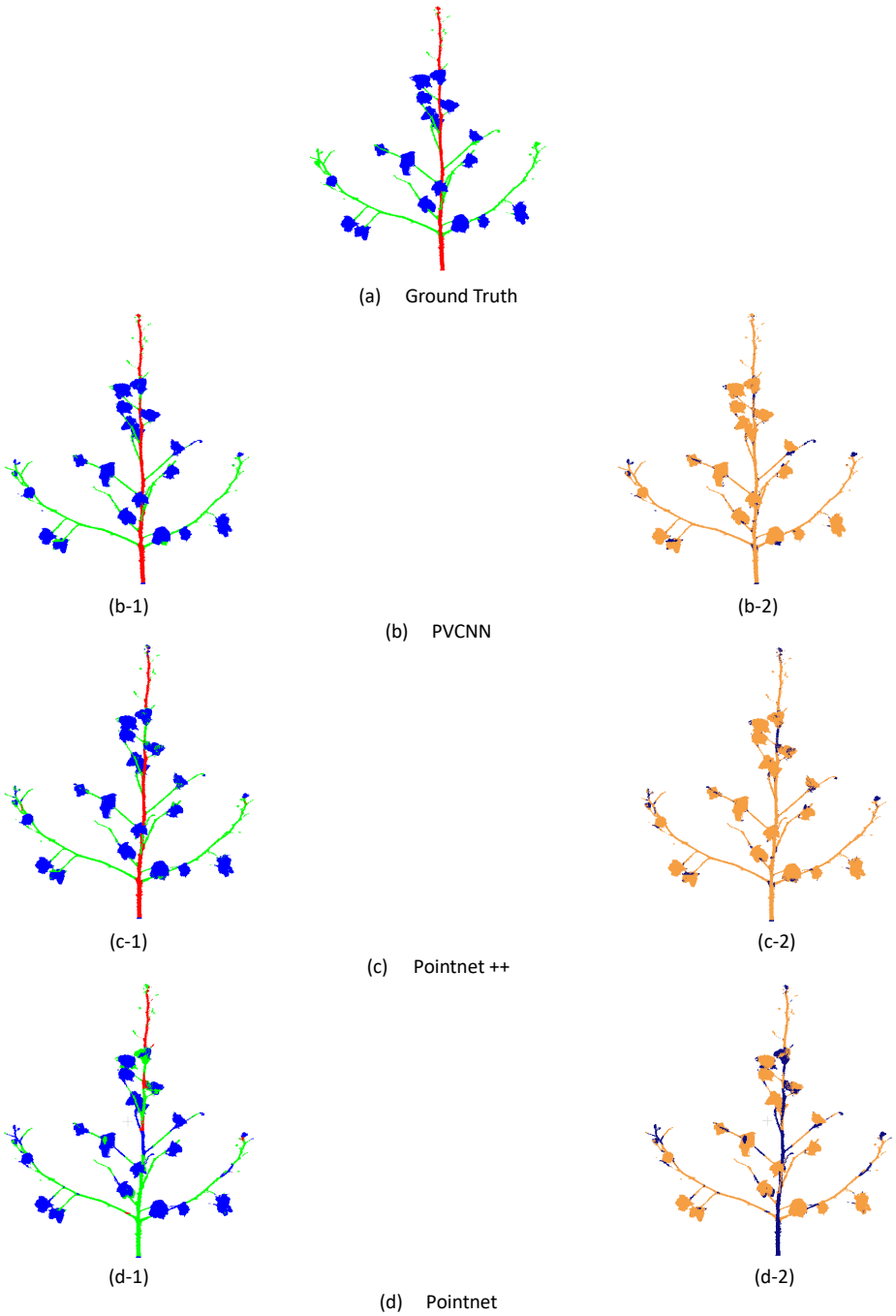

Figure 5: Plant organ segmentation results. Figure (a) represents the ground truth segments. The red, green, and blue represent main stem, branch, and cotton bolls respectively. Figures (b-1) (c-1) (d-1) represents the segments predicted by PVCNN, Pointnet++, Pointnet, respectively. The red, green, and blue are the points predicted as main stem, branch, and bolls respectively. Figures (b-2) (c-2) (d-2) represent the correctness of predicted points. The correctly and incorrectly predicted points are colored in orange and navy-blue, respectively.

Since the network operates on sub-bocks of point clouds and not on the point cloud as a whole, it shows mis-prediction in some parts of the branch as a main stem (Figure 6(b)). In Figure 6(b), it can be observed that the branches that are almost parallel to the *z*-axis are predicted as main stems generally. Similarly, some parts of main stems were also classified as branches (Figure 6(a)), since the majority of the training data consisted of samples in which the main stem was almost vertical. This misclassification is prominent in cases where parts of the main stem are not vertical and happen to deviate at a significant angle from *z*-axis. (Figure 6(c)) shows the pointcloud where more than 50% of the main stem is slanted at an angle and is predicted as a branch. Another misprediction is in the case of branch parts which are near the cotton bolls. (Figure 6(d)) The branch parts near the cotton bolls are also predicted as cotton. This is because the training data has similar annotation errors because of manual labelling. Because of human error, the branch parts attached to the cotton bolls are also labelled as cotton bolls in the ground truth. Finally, one of test point clouds showed the misclassification of lower main stem parts as cotton bolls (Figure 6(d)). This is because the cotton bolls have generally a higher thickness compared with the main stem. The main stem is quite thin in most of the point clouds in the dataset.

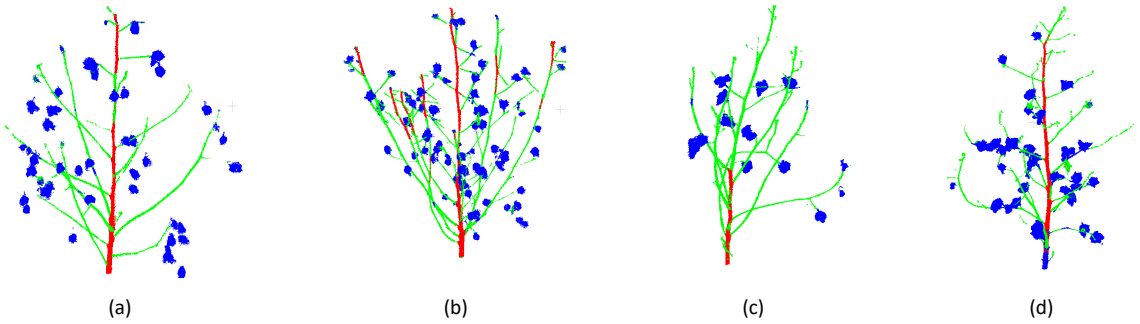

Figure 6: Examples to erroneous predictions of PVCNN

For improving the segmentation results, the post processing is applied on predicted labels. Post-processing was performed to correct the main stem parts predicted as branches (Figure 6(a)) and the branch parts predicted as main stem (Figure 6(b)). As can be seen in the Figure 6(a), small parts in the main stem are marked green by the network in the test data. To apply postprocessing to correct this, the operation is started from the lowest slice of points predicted as main stem. The slice is propagated upwards along z-axis to see if there are any non-red points in the region to be marked as red. After applying the post-processing on the mainstem, the remaining points are observed to convert any remaining red points to green. In this way, the branch parts predicted as main stems are corrected. (Figure 7(a-2)). The segmentation results before and after the postprocessing are given in Figure 7. This postprocessing corrects the majority of main stem mispredictions in the test data but not with main stems with significantly slanted orientation (Figure 6(c)).

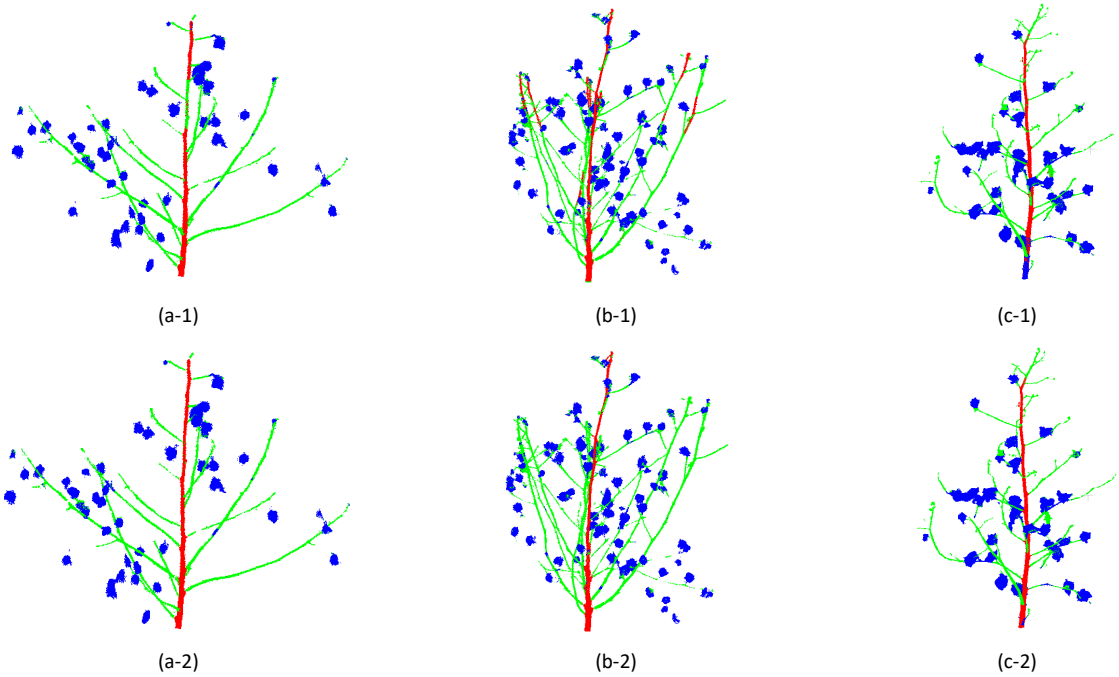

Figure 7: Postprocessing results applied to plant organs segmented using PVCNN. Top row shows the segmentation results achieved from PVCNN. Bottom row shows the results achieved after postprocessing.

Trait Extraction

The postprocessed results of test data are used in the extraction of phenotypic traits. Four phenotypic, traits including main stem diameter, main stem height, number of nodes, and average internode distance, of cotton plants were extracted and analyzed. The main stem traits extraction as in (Figure 8) is performed by first filtering the points in the test data to include only main stem. The main stem points then are used to extract the height and diameter of the main stem. For estimating the diameter, the points in the lowest 1cm region of main stem (Figure 8(b), 8(c)) are taken and mapped to the x-y plane (Figure 8(d)). A circle in the x-y plane is detected using circle fitting method. The detected circle (as shown in Figure 8(d)) is used to calculate the diameter of main stem. The height of the main stem is estimated as the difference in the maximum and minimum value on the *z*-axis of the main stem points.

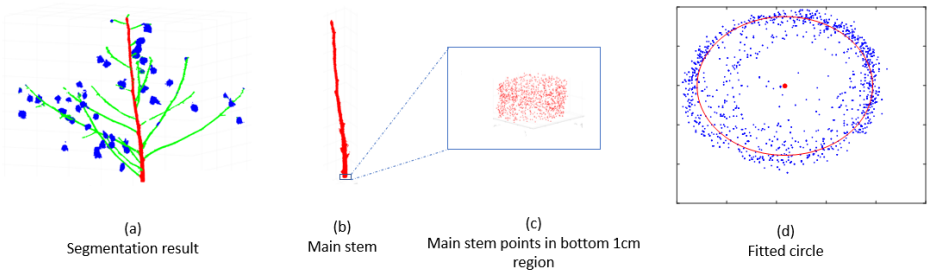

Figure 8: Mainstem height and diameter extraction. Non red points are removed. Points in lower 1cm part are selected. A circle is fitted to the selected points.

Postprocessed results were used to extract the nodes in the cotton plant (Figure). This is achieved by detecting the parts of the main stem where it is connected with some branch points (green points). Each of those parts are identified as nodes. If the two nodes are less than 2 cm apart, they are averaged to form a single node. The detected nodes of a test point cloud are shown in (Figure 9). Finally, the distance between each node was computed and averaged to calculate the average internode distance of the plant. The methods used for the trait extraction of postprocessed results are also used in the trait extraction of ground truth annotations.

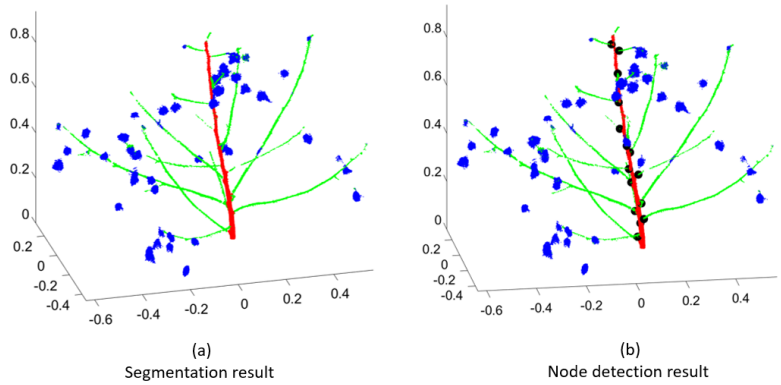

Figure 9: Node detection in segmented plant organs. The left figure shows the plant organ segmentation obtained from PVCNN. The right figure shows the nodes detected in black circles on the main stem.

Linear regression was performed to analyze the trait extraction results of ground truth and postprocessed segmentation results. (Figure10) show the analysis of main stem diameter, main stem height, number of nodes and average internode distance for predicted and ground truth of test data. On the main stem diameter, the linear regression shows the R-squared value of 0.9 (Figure 10(a)). On main stem height and average internode distance, a root mean square error (RMSE) of less than 0.3 meters is achieved (Figure 10(b), 10(d)). The RMSE in number of nodes of cotton plants is lower than 3 (Figure 10(c)). The linear regression of the main stem height and number of nodes shows the presence of an outlier. This outlier corresponds to an unusual plant architecture in our dataset having a slanted main stem. As shown in Figure 6(c), the PVCNN is able to predict only less than half of the main stem correctly. Hence, on the trait extraction, this sample shows as an outlier (Figure 10(b) & 10(c)). For this sample, the number of nodes was predicted to be 11 compared to 19 as the ground truth(Figure 10(c)). Similarly, the main stem height is predicted to be less than 0.5m compared to more than 1.2m. After excluding the outlier, the RMSE is reduced to less than 5cm for main stem height (Figure 11(a)) and less than 2 for number nodes (Figure 11(c)). As a result, R-squared value of around 0.9 and 0.6 is achieved for main stem height and number of nodes, respectively (Figure 11(a) & 11(b)). On average internode distance, the RMSE of less than 1cm is achieved.

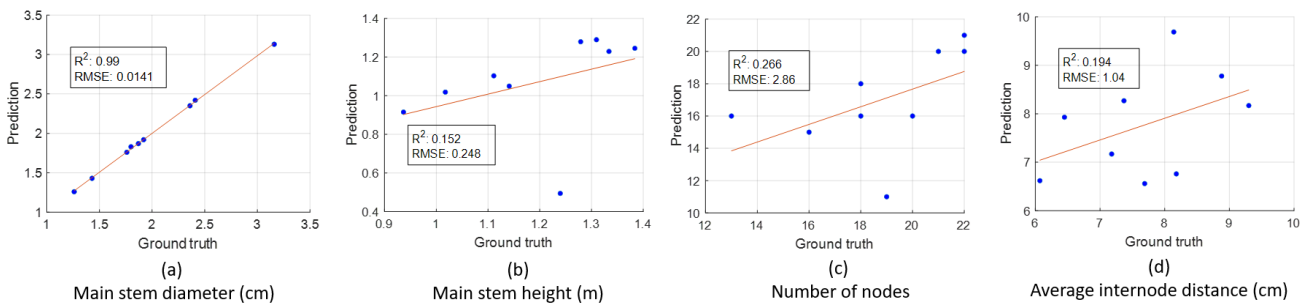

Figure 10: Linear regression of phenotypic traits between ground truth and predictions

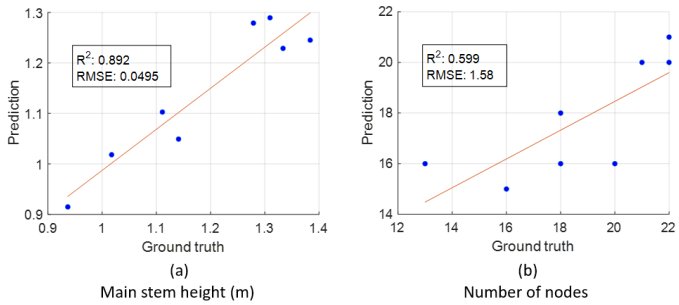

Figure 11: Linear regression on main stem height and number of nodes between ground truth and predictions after removing outlier.

Conclusions

3D data provides more accurate representation as compared to 2D images. The segmentation using Point-voxel CNN was performed to leverage both point and voxel representation of input data. The experiments showed that the PVCNN outperformed its baselines in terms of precision, recall, accuracy, IOU, and mIOU. The postprocessing was able to correct the main stem and branches in most of the predicted results. The phenotypic traits including main stem height, main stem diameter, number of nodes and average internode distance were extracted on the postprocessed results with satisfactory results. This plant organ segmentation method based on a 3D point-voxel CNN makes it possible to measure phenotypic traits related to plant architecture, which is important for plant breeding and physiology.

References

Atefi, A., Y. Ge, S. Pitla and J. Schnable (2019). "In vivo human-like robotic phenotyping of leaf traits in maize and sorghum in greenhouse." Computers and Electronics in Agriculture 163: 104854. https://doi.org/10.1016/j.compag.2019.104854

Bai, G., Y. Ge, W. Hussain, P. S. Baenziger and G. Graef (2016). "A multi-sensor system for high throughput field phenotyping in soybean and wheat breeding." Computers and Electronics in Agriculture 128: 181-192. https://doi.org/10.1016/j.compag.2016.08.021

Diaz-Garcia, L., G. Covarrubias-Pazaran, B. Schlautman, E. Grygleski and J. Zalapa (2018). "Image-based phenotyping for identification of QTL determining fruit shape and size in American cranberry (Vaccinium macrocarpon L.)." PeerJ 6: e5461. https://doi.org/10.7717/peerj.5461

Dutagaci, H., P. Rasti, G. Galopin and D. Rousseau (2020). "ROSE-X: an annotated data set for evaluation of 3D plant organ segmentation methods." Plant methods 16(1): 1-14. https://doi.org/10.1186/s13007-020-00573-w

Elnashef, B., S. Filin and R. N. Lati (2019). "Tensor-based classification and segmentation of three-dimensional point clouds for organ-level plant phenotyping and growth analysis." Computers and electronics in agriculture 156: 51-61. https://doi.org/10.1016/j.compag.2018.10.036

Herrero-Huerta, M., R. Lindenbergh and W. Gard (2018). "Leaf movements of indoor plants monitored by terrestrial LiDAR." Frontiers in plant science 9: 189. https://doi.org/10.3389/fpls.2018.00189

Jay, S., G. Rabatel, X. Hadoux, D. Moura and N. Gorretta (2015). "In-field crop row phenotyping from 3D modeling performed using Structure from Motion." Computers and Electronics in Agriculture 110: 70-77. https://doi.org/10.1016/j.compag.2014.09.021

Le, T. and Y. Duan (2018). Pointgrid: A deep network for 3d shape understanding. Proceedings of the IEEE conference on computer vision and pattern recognition. https://doi.org/10.1109/CVPR.2018.00959

Li, M., L. L. Klein, K. E. Duncan, N. Jiang, J. P. Londo, A. J. Miller and C. N. Topp (2019). "Characterizing grapevine (Vitis spp.) inflorescence architecture using X-ray imaging: implications for understanding cluster density." bioRxiv: 557819. https://doi.org/10.1093/jxb/erz394

Li, Y., R. Bu, M. Sun, W. Wu, X. Di and B. Chen (2018). "Pointcnn: Convolution on x-transformed points." Advances in neural information processing systems 31: 820-830.

Liu, Z., H. Tang, Y. Lin and S. Han (2019). "Point-voxel cnn for efficient 3d deep learning." arXiv preprint arXiv:1907.03739.

Mairhofer, S., S. Zappala, S. R. Tracy, C. Sturrock, M. Bennett, S. J. Mooney and T. Pridmore (2012). "RooTrak: automated recovery of three-dimensional plant root architecture in soil from X-ray microcomputed tomography images using visual tracking." Plant physiology 158(2): 561-569. DOI: https://doi.org/10.1104/pp.111.186221

Metzner, R., A. Eggert, D. van Dusschoten, D. Pflugfelder, S. Gerth, U. Schurr, N. Uhlmann and S. Jahnke (2015). "Direct comparison of MRI and X-ray CT technologies for 3D imaging of root systems in soil: potential and challenges for root trait quantification." Plant methods 11(1): 1-11. https://doi.org/10.1186/s13007-015-0060-z

Omasa, K., F. Hosoi and A. Konishi (2007). "3D lidar imaging for detecting and understanding plant responses and canopy structure." Journal of experimental botany 58(4): 881-898. https://doi.org/10.1093/jxb/erl142

Panjvani, K., A. V. Dinh and K. A. Wahid (2019). "LiDARPheno–A low-cost lidar-based 3D scanning system for leaf morphological trait

extraction." Frontiers in plant science 10: 147. https://doi.org/10.3389/fpls.2019.00147

Polder, G., P. M. Blok, H. A. de Villiers, J. M. van der Wolf and J. Kamp (2019). "Potato virus Y detection in seed potatoes using deep learning on hyperspectral images." Frontiers in plant science 10: 209. https://doi.org/10.3389/fpls.2019.00209

Qi, C. R., H. Su, K. Mo and L. J. Guibas (2017). Pointnet: Deep learning on point sets for 3d classification and segmentation. Proceedings of the IEEE conference on computer vision and pattern recognition. https://doi.org/10.1109/CVPR.2017.16

Qi, C. R., L. Yi, H. Su and L. J. Guibas (2017). "Pointnet++: Deep hierarchical feature learning on point sets in a metric space." arXiv preprint arXiv:1706.02413.

Shi, W., R. van de Zedde, H. Jiang and G. Kootstra (2019). "Plant-part segmentation using deep learning and multi-view vision." Biosystems Engineering 187: 81-95. https://doi.org/10.1016/j.biosystemseng.2019.08.014

Sun, S., C. Li, P. W. Chee, A. H. Paterson, Y. Jiang, R. Xu, J. S. Robertson, J. Adhikari and T. Shehzad (2020). "Three-dimensional photogrammetric mapping of cotton bolls in situ based on point cloud segmentation and clustering." ISPRS Journal of Photogrammetry and Remote Sensing 160: 195-207. https://doi.org/10.1016/j.isprsjprs.2019.12.011

Sun, S., C. Li, A. Paterson and P. Chee (2020). Three-dimensional cotton plant shoot architecture segmentation and phenotypic trait characterization using terrestrial LiDAR point cloud data. 2020 ASABE Annual International Virtual Meeting, American Society of Agricultural and Biological Engineers. https://doi.org/10.13031/aim.202001267

Sun, S., C. Li, A. H. Paterson, Y. Jiang, R. Xu, J. S. Robertson, J. L. Snider and P. W. Chee (2018). "In-field high throughput phenotyping and cotton plant growth analysis using LiDAR." Frontiers in plant science 9: 16. https://doi.org/10.3389/fpls.2018.00016

Tracy, S. R., C. R. Black, J. A. Roberts, A. McNeill, R. Davidson, M. Tester, M. Samec, D. Korošak, C. Sturrock and S. J. Mooney (2012). "Quantifying the effect of soil compaction on three varieties of wheat (Triticum aestivum L.) using X-ray Micro Computed Tomography (CT)." Plant and Soil 353(1): 195-208. https://doi.org/10.1007/s11104-011-1022-5

Voora, V., C. Larrea and S. Bermudez (2020). Global Market Report: Cotton, JSTOR.

Wang, P.-S., Y. Liu, Y.-X. Guo, C.-Y. Sun and X. Tong (2017). "O-cnn: Octree-based convolutional neural networks for 3d shape analysis." ACM Transactions on Graphics (TOG) 36(4): 1-11. https://doi.org/10.1145/3072959.3073608

Wang, Y., Y. Sun, Z. Liu, S. E. Sarma, M. M. Bronstein and J. M. Solomon (2019). "Dynamic graph cnn for learning on point clouds." Acm Transactions On Graphics (tog) 38(5): 1-12. https://doi.org/10.1145/3326362

Wang, Z. and F. Lu (2019). "VoxSegNet: Volumetric CNNs for semantic part segmentation of 3D shapes." IEEE transactions on visualization and computer graphics 26(9): 2919-2930. https://doi.org /10.1109/tvcg.2019.2896310

Zhang, Z., B.-S. Hua, D. W. Rosen and S.-K. Yeung (2019). Rotation invariant convolutions for 3d point clouds deep learning. 2019 International Conference on 3D Vision (3DV), IEEE. https://doi.org/10.1109/3DV.2019.00031

Zhang, Z., B.-S. Hua and S.-K. Yeung (2019). Shellnet: Efficient point cloud convolutional neural networks using concentric shells statistics. Proceedings of the IEEE/CVF International Conference on Computer Vision. https://doi.org/10.1109/ICCV.2019.00169

Zhou, Y. and O. Tuzel (2018). Voxelnet: End-to-end learning for point cloud based 3d object detection. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. https://doi.org/ 10.1109/CVPR.2018.00472

Zhu, H., B. Chu, C. Zhang, F. Liu, L. Jiang and Y. He (2017). "Hyperspectral imaging for presymptomatic detection of tobacco disease with successive projections algorithm and machine-learning classifiers." Scientific reports 7(1): 1-12. https://doi.org/10.1038/s41598-017-04501-2

Ziamtsov, I. and S. Navlakha (2019). "Machine learning approaches to improve three basic plant phenotyping tasks using three-dimensional point clouds." Plant physiology 181(4): 1425-1440. https://doi.org/10.1104/pp.19.00524