2024 Systems and Information Engineering Design Symposium (SIEDS) | 979-8-3503-8514-4/24/$31.00 ©2024 IEEE | DOI: 10.1109/SIEDS61124.2024.10534633

Plant Organelle Segmentation using 3D Point Clouds and PointNet

Rony Dahdal[1], Blake Hament[1]

[1] Elon University, Elon, NC 27244 USA (e-mail: rdahdal2@elon.edu, bhament@elon.edu).

***Abstract*— Monitoring crops is an essential task in maintaining the life of plants. State-of-the-art agricultural systems are capable of capturing 3D data of individual crops and reconstructing digital twins. A prominent issue in the field is extracting information about the subject plant from the point cloud data, particularly classifying organelle. To address this problem, a deep neural network with a PointNet-like architecture is proposed to identify organelle of the subject plant. Using datasets of plant images taken from an RGB-camera attached to a robotic arm, 3D point cloud data is generated with structure-from-motion photogrammetry techniques, which is used to train a network to segment local regions of the model. A proof-of-concept segmentation network is presented, trained using a large multi-view dataset of avocados images to identify organelle such as the seed, interior pulp, and exterior skin. The results collated evaluate the model throughout training using metrics such as RMSE of training and validation datasets.**

I. INTRODUCTION

Previous research conducted in environmental engineering and precision agriculture has developed autonomous systems to monitor crop development, using image sensors and structure-from-motion algorithms to reconstruct and capture the growth of crops [1]. However, these methods lack a focus on individual crop development, leading to data of parts of plants suffering from occlusion [2]. To address issues with plant data occlusion, our work seeks to introduce an autonomous crop-monitoring system that uses a robotic arm to aid in fostering and monitoring plant development. To properly monitor the growing crop, our system must develop a level of understanding of reconstructed 3D point cloud data (PCD), particularly the ability to comprehend local geometry and segment plant anatomy. Understanding of plant anatomy would enable systems to properly evaluate and tend to the development of growing crops. Previous work has introduced supervised learning techniques like Random Forests [3], and graph-based approaches [4] to segment reconstructed PCD. However, recent advancements in 3D deep learning techniques allow neural networks to learn segmentation tasks directly from raw PCD. To interpret the reconstructed 3D model, we employ PointNet [5], a deep learning framework, to segment the organelle of the crop, which would provide our system with an understanding of plant anatomy and act accordingly in response to its growth over time. This study presents a proof-of-concept implementation of PointNet to segment 3D point cloud representations beginning with “toy problem” experiments focused on reconstructed multi-view data of avocados. Our results suggest that PointNet-like architectures are sufficient in understanding and classifying anatomy of 3D plant data.

II. PREVIOUS WORK

A. PointNet

PointNet is a deep neural network architecture that learns to interpret unordered PCD. PointNet is able to accurately classify and segment 3D point clouds through its architecture based on multi-layer perceptions, which process individual points, and a max-pooling layer that aggregates learned information throughout all points in the point cloud to extract global features.

III. METHODS

A. MVImgNet

For sufficient and high-quality avocado PCD to investigate our toy problem, we turn to MVImgNet [6], a large-scale dataset of multi-view images with over 87,000 point cloud samples from 150 object categories, generated through dense reconstruction of 6.5 million frames. MVImgNet included over 700 samples of avocado PCD which was implemented into the PointNet segmentation training pipeline.

B. Segments.AI

The PointNet segmentation network requires preprocessed samples that are assigned corresponding labels. To achieve this and prepare our avocado PCD for training, we manually labeled the anatomy of the avocado, utilizing

Segments.AI's data labeling platform to assign ground-truth values to each individual point of each avocado reconstructed model. Our labels consisted of pulp, the interior of the avocado, skin, the exterior layer of the avocado, and seed, the center of the avocado. In accordance with Segments.AI's platform, these labels were enumerated accordingly:

- Pulp: value of 1.
- Skin: value of 2.
- Seed: value of 3.

C. PointNet Segmentation Network

PointNet's segmentation network requires a combination of local and global understanding of its learned PCD. After PointNet computes its global point cloud feature vector through its architecture, the global feature is concatenated with local point features, which in our case is the corresponding class the point belongs to (pulp, skin, seed). This addition of global understanding creates new per point features that are conscious of both local and global information, enabling robust segmentation, which relies on local geometry. Our segmentation model uses a learning decay schedule that starts with an initial learning rate of 0.003 and decays in half every 5 epochs, or 19 steps in our case, for a total of 760 learning steps (Figure 1) and uses the cross-entropy loss function which is ideal for classification and segmentation tasks [7].

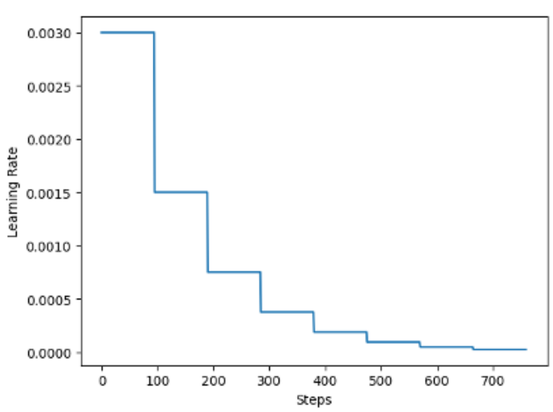


Figure 1. Learning rate schedule for the segmentation model that decays initial learning rate of 0.003 by 0.5 for every 19 steps, for a total of 760 total training steps.

D. Data Preprocessing

Data that is classified as labeled is used as training and validation data for our segmentation network. In this paper, we consider a dataset of 24 avocado point clouds that were manually labeled using Segments.AI. PointNet requires a fixed number of points that is identical across every training sample, for which we chose 1024 number of points, as this is the dimension of global features learned by the model. We randomly sample 1024 points from each avocado point cloud and normalize to ensure data is scale-invariant. Additionally, we undersample the number of points in the majority class of our data, pulp (Figure 3). Mitigating data-imbalance prior to training PointNet is beneficial to increase minority class prediction and decreasing chance of overfitting the majority class [8]. Finally, we apply random shuffling and jitter to our data before feeding it into the training pipeline (Figure 2) to ensure general understanding of the point cloud geometry.

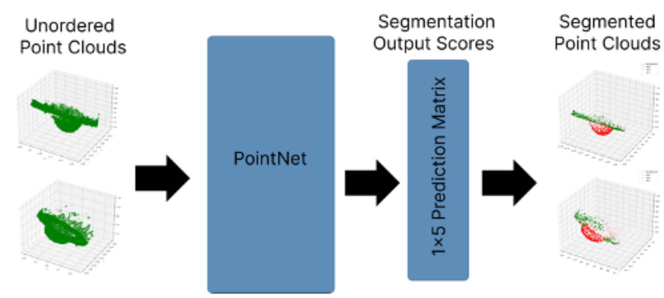


Figure 2. Schematic of training pipeline, beginning with unordered, dense PCD, that is used for training PointNet prior to normalization and fixed number sampling. Shown right is the normalized and predicted labeled segmentation of our model on corresponding training samples.

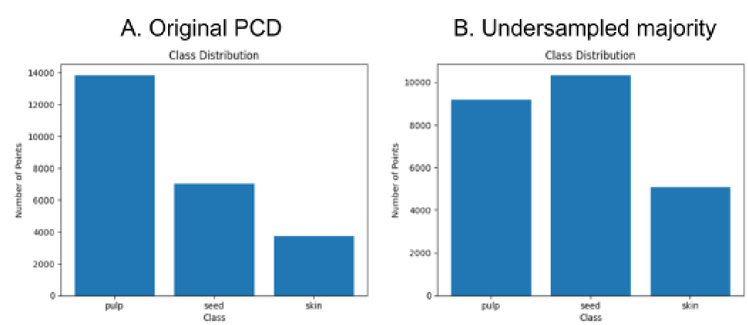


Figure 3. A) Original class distribution with imbalance in number of pulp points in PCD, which causes overfitting of pulp class. B) Post-undersampling class distribution that mitigates imbalance in PCD.

E. TensorFlow and Keras

Using machine learning Python packages such as TensorFlow and Keras [9], we are able to incorporate PointNet model layers and functions, such as convolutional and multi-layer perceptron blocks, as well as transformation networks, which are integral for PointNet's architecture. We used Keras to run experiments with our model, incorporate the cross-entropy loss function, and estimate high-performance hyperparameters to train our model with a grid search that tests different learning rates and batch sizes across 40 epochs.

IV. RESULTS

Our segmentation network achieved a favorable accuracy of 64% on a validation dataset of 5 avocado point clouds that depicts gained knowledge of local shapes.

TABLE I. PointNet Segmentation Network Accuracy

PointNet Segmentation Network Accuracy		
Epoch Count	***Training accuracy***	***Validation accuracy***
25	89%	64%
30	91%	63%
40	93%	57%

The model begins to show signs of overfitting at 40 epochs, as training accuracy and validation loss increase (Fig. 4A) while validation accuracy decreases (Fig. 4B), which infers the model grew too reliant on its training data. With fewer epochs that avoid overfitting, PointNet proved sufficient as a method of understanding and differentiating plant anatomy (Fig. 5) in the scope of our toy problem and is promising for future implementation for segmenting 3D plant data collected from our autonomous individual plant farming system.

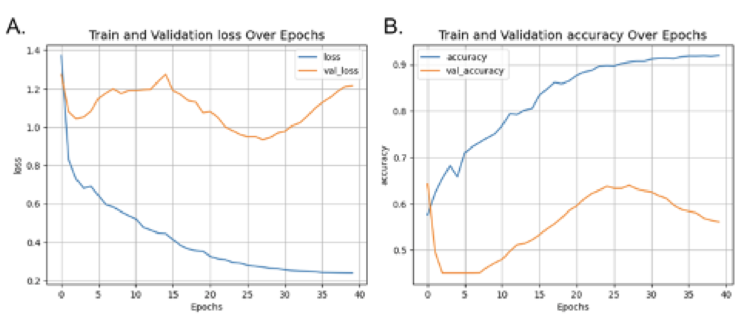


Figure 4. A) Segmentation network training and validation accuracy curves over 40 epochs. B) Training and validation loss curves over 40 epochs.

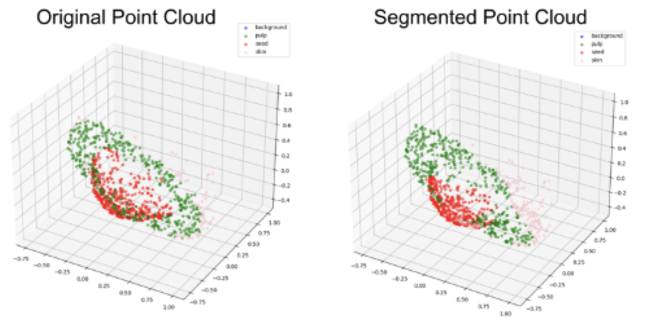


Figure 5. Sample validation point cloud with its ground-truth labels (shown left) and predicted labels from our PointNet segmentation network (shown right). The segmented point cloud displayed PointNet's learned local understanding of avocado organelle.

V. DISCUSSION

We selected 24 samples of avocado PCD that clearly depicted local organelle regions in order to validate our model's ability to differentiate between complex shapes, such as the pear-shaped pulp, thin layer of skin, and spherical seed of the avocado point clouds.

For every point cloud sample, after each individual point was manually labeled into one of the three class categories (pulp, skin, seed), each sample was one-hot encoded. We used an 80/20 training and validation split of our dataset: 19 samples for training, and 5 samples for validation. For its loss function, our PointNet model used categorical cross-entropy for its robustness in classification tasks, such as part segmentation. Prior to training with PointNet, we performed three important transformations to our PCD: 1) fixed point sampling which was followed by 2) mean normalization and 3) random jitter. PointNet architecture requires identical lengths of point cloud samples. To satisfy this requirement, we employ fixed point sampling to randomly select a fixed number of 1024 points from each sample point cloud. Afterwards, we normalize the point clouds by subtracting the original point cloud values from its mean and then dividing by the maximum Euclidean norm using the Numpy package, which ensures the maximum length of any vector in the point cloud to become 1. Finally, in order to apply perturbation and jitter to improve classification and avoid overfitting [10], we apply noise to our training dataset by generating random values between 0.001 and -0.001 with the same shape as our original point cloud, which is then added to our original point cloud to introduce variation into the data.

While training, we noticed a significant imbalance of data between our three class categories, in which pulp made up the majority of points in our PCD (Figure 3A). This caused overfitting to occur while training our model, leading PointNet to predict every point in a validation point cloud as pulp in order to achieve high accuracy. This imbalance limited PointNet's understanding of local regions by causing it to assume pulp made up both global and local geometry in each point cloud. We mitigated this imbalance by undersampling pulp, retaining 55% of its original size (Figure 3B). Additionally, we undersampled seed and decreased its size by 5% in order to avoid overfitting of seed and ensure PointNet learns the minority class, skin.

To evaluate the combination of hyperparameters with the highest performance on the validation set, we used a grid search. Our model was trained across 40 epochs exploring all possible combinations of batch sizes (1, 2, and 4), and initial learning rates (0.001 and 0.003). After exploring the hyperparameter space, we identified a batch size of 1 and an initial learning rate of 0.003 achieved the highest validation accuracy of 64% (Table I). While our model achieved a training accuracy of 89% at its 25th epoch, subsequent epochs exhibited signs of overfitting to its training data, as training accuracy increased, but validation loss and accuracy worsened. These findings are illustrated in training and validation accuracy curves (Figure 4A) and loss curves (Figure 4B) indicating the model overtrained and began to perform poorly on previously unseen PCD beyond 25 epochs.

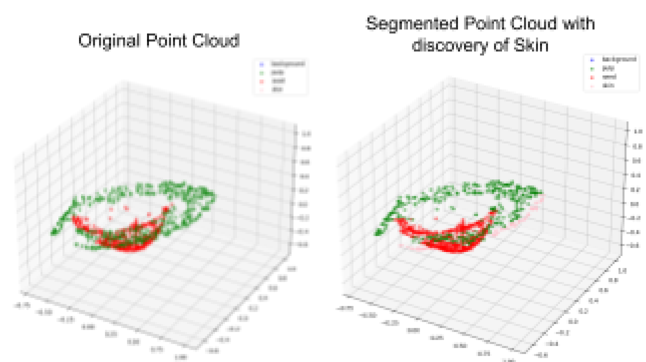
Our results in solving this toy problem are encouraging to the potential of PointNet to inform systems of local anatomy displayed in unordered 3D data. The model, trained on a miniscule dataset that consisted of 19 samples, gained an ability to understand general geometric shapes exhibited in a typical avocado, such as the spherical, centered seed (Figure 5). Additionally, one motivating result we discovered was PointNet's stability and robustness to identify local geometry across its training dataset, regardless of missing labels. For instance, when tested on a training sample that didn't include labels of skin, PointNet was still able to predict the thin, exterior geometry that made up the avocado skin category, along with the shape of the pulp and seed accurately (Appendix I). This paper's findings encourage that PointNet and PointNet-like architecture are able to properly process unordered plant PCD and generalize its understanding of plant anatomy.

PointNet's reliance on global feature vectors and overall global understanding leads to a lack of local structure information and a limited capacity to capture detailed features that would assist in challenging tasks such as part segmentation [11]. To address these limitations, researchers have developed improved architectures such as PointNet++ [12], which introduces hierarchical feature learning: an approach that more effectively captures local relationships and proves greater proficiency in understanding more complex shapes. In the future, we look to further improve our approach to understanding plant anatomy by using cutting-edge hierarchical feature learning structures such as PointNet++, as well as experimenting with state-of-the-art performing pretrained models developed by previous research, such as PointNeXt [13], that are well-generalized for downstream tasks such as part segmentation [14].

VI. CONCLUSION

In this work, we propose a proof-of-concept solution for plant part segmentation with "toy problem" experiments on avocado point cloud data that proves sufficient in understanding local point cloud geometry and classifying organelle. We provide an evaluation metric for our approach and suggest future directions to increase performance. We will further improve the accuracy of our model with cutting-edge pre-trained PointNet models and state-of-the-art performing models.

VI. APPENDIX



Appendix I. Sample training point cloud (shown left) without skin labels our PointNet segmentation model's segmentation prediction (shown right) exhibiting discovery of skin class. This is encouraging to PointNet's capability to understand the subtle, exterior geometry of plant anatomy.

REFERENCES

[1] J. Dong, J. G. Burnham, B. Boots, G. Rains, and F. Dellaert, "4D crop monitoring: Spatio-temporal reconstruction for agriculture," 2017 IEEE International

Conference on Robotics and Automation (ICRA), Singapore, 2017, pp. 3878-3885.
[2] F. Saeed, S. Sun, J. Rodriguez-Sanchez, J. Snider, T. Lie, and C. Li, “Cotton plant part 3D segmentation and architectural trait extraction using point voxel convolutional neural networks,” Plant methods, vol. 19, Mar. 2023, pp. 33.
[3] J. Niemeyer, F. Rottensteiner and U. Soergel, "Classification of urban LiDAR data using conditional random field and random forests," Joint Urban Remote Sensing Event 2013, Sao Paulo, Brazil, 2013, pp. 139-142.
[4] K. Mirande, C. Godin, M. Tisserand, J. Charlaix, F. Besnard, and F. Hétroy-Wheeler, “a graph-based approach for simultaneous semantic and instance segmentation of plant 3D point clouds,” Front. Plant Sci, vol. 13, Nov. 2022.
[5] C. Qi, H. Su, K. Mo, and L. Guibas, “PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation,” 2017. [Online]. Available: arXiv:1612.00593.
[6] X. Yu, M. Xu, Y. Zhang, H. Liu, C. Ye, Y. Wu, et al, “MVImgNet: A Large-scale Dataset of Multi-view Images,” 2023. [Online]. Available: arXiv:2303.06042.
[7] D. Kline and V. Berard, “Revisiting squared-error and cross-entropy functions for training neural network classifiers,” Neural Computing & Applications, vol. 14, 2023, pp. 310–318.
[8] H. Lin and M. Nguyen, “Boosting Minority Class Prediction on Imbalanced Point Cloud Data,” Appl. Sci, vol. 10, 2020, pp. 973.
[9] François Chollet. “Keras,” 2015.
[10] R. Zur, Y. Jiang, L. Pesce, and K. Drukker, “Noise injection for training artificial neural networks: A comparison with weight decay and early stopping,” Medical Physic, vol. 36, Oct. 2009, pp. 4810–18.
[11] S. Bello, C. Wang, N. Wambugu, and J. Adam, “FFPointNet: Local and global fused feature for 3D point clouds analysis,” Neurocomputing, vol. 461, Oct. 2021, pp. 55-62.
[12] C. Qi, L. Yi, H. Su, and L. Guibas, “PointNet++: Deep Hierarchical Feature Learning on Point Sets in a Metric Space,” 2017. [Online]. Available: arXiv:1706.02413.
[13] G. Qian, Y. Li, H. Peng, J. Mai, H. Hammoud, et al. “PointNeXt: Revisiting PointNet++ with Improved Training and Scaling Strategies,” 2022. [Online]. Available: arXiv:2206.04670.
[14] S. Xie, J. Gu, D. Guo, C. Qi, L. Guibas, and O. Litany, “PointContrast: Unsupervised Pre-training for 3D Point Cloud Understanding,” 2020. [Online]. Available: arXiv:2007.10985.