Segmentation and 3D reconstruction of root systems and *Humulus lupulus* for high-throughput phenotyping using deep neural networks and 3D imaging

by

Mary Elizabeth Cassity

A thesis submitted to the Graduate Faculty of
Auburn University
in partial fulfillment of the
requirements for the Degree of
Master of Science

Auburn, Alabama
August 05, 2023

Keywords: Root system architecture, X-ray computed tomography, Instance segmentation, DBSCAN, *Humulus lupulus,* Stereo vision, Semantic segmentation, Point clouds

Approved by

Yin Bao, Chair, Assistant Professor of Biosystems Engineering
Paul C. Bartley, Assistant Professor of Horticulture
Tanzeel Rehman, Assistant Professor of Biosystems Engineering
Alvaro Sanz-Saez, Assistant Professor of Crop, Soil and Environmental Sciences

Abstract

The availability of artificial intelligence (AI) tools for computer vision applications has greatly increased in recent years, with many advanced tools available to researchers across various fields. Precision agriculture has likewise seen an increase in the application of computer vision technologies to the field, including the application of computer vision to crop phenotyping. AI-empowered phenotyping can significantly reduce labor burdens and risks, providing efficient and high-throughput tools to process numerous plants efficiently. This paper explores two applications of computer vision to challenges in horticultural phenotyping in three-dimensional (3D) space.

The first work addresses the segmentation and reconstruction of root systems from X-ray computed tomography (CT) images. Accurate phenotyping of root system architecture (RSA) is a significant challenge in horticultural phenotyping because the root system is below the soil, occluding it from view and requiring traditional phenotyping techniques to remove the root system from the soil. Segmentation and reconstruction of the root system from X-ray CT images is therefore integral to observing the root system undisturbed in the soil and over time in response to abiotic and biotic stress. Numerous image-processing techniques have been applied to the problem of segmentation of below soil root systems from CT images. However, these methods often require user intervention and require input parameters tuned to plant species and soil conditions. A recent deep learning approach employed a volumetric encoder-decoder to achieve high scores for common computer vision accuracy metrics. However, training a volumetric model relies on copious amounts of hand-annotated training data. We propose using a

two-dimensional (2D) Mask R-CNN model for instance segmentation of root cross sections in CT images. The 2D predictions can be merged to produce a 3D prediction. We also propose an automated parameter tuning pipeline for density-based spatial clustering of applications with noise (DBSCAN) to remove noise from the 3D segmentation. The proposed method was evaluated on scans of poinsettias and onions and achieved average scores of 0.734, 0.868, 0.749, and 0.669 for precision, recall, dice, and IoU, respectively, utilizing only 1% of the training dataset. Our proposed method is resource efficient, capitalizing on the training efficiency of a 2D model as well as 3D information during the unsupervised clustering using DBSCAN.

Our second work addresses the use of stereo vision for 3D reconstruction of *Humulus lupulus* (hops) as well as a semi-automated pipeline for phenotypic trait extraction of vine length, leaf area, and biomass. High-throughput computer vision tools for phenotyping are important for variety trials of hops, as they facilitate fast and safe measurement taking as well as non-destructive measurement of leaf area and biomass. However, studies developing computer vision and machine learning for morphological phenotyping are not common for vine plants, and even less work has been completed for hops. Therefore, this work will develop and evaluate a method for hop morphological phenotyping. A 2D transformer, SegFormer, was trained for semantic segmentation of the hops and used to segment hop plants from 3D point cloud scenes retrieved from a ZED 2 stereo camera. Measurements of vine length, leaf area, and biomass were derived from the segmented point clouds, yielding high R^2 values of 0.79, 0.95, and 0.91, respectively, indicative of a strong correlation between the derived measurements and the ground truth measurements.

Acknowledgments

First, I would like to thank my advisor, Dr. Yin Bao, for giving me the opportunity to pursue and complete a MS. He is truly a great and patient advisor. Furthermore, I would like to thank my committee members, Dr. Paul C. Bartley, Dr. Tanzeel Rehman, and Dr. Alvaro Sanz-Saez, for their contributions of time, expertise, and feedback, and for the encouragement they gave me. I would also like to thank Dr. Oladiran Fasina for suggesting I speak with Dr. Bao during my junior year of undergraduate studies about furthering my education through a MS.

I would like to thank my lab mates, Kamand Bagherian, Rafael Bidese, Preston Langston, Nariman Niknejad, Spencer Overton, Sharif Shabani, and Puranjit Singh for both their collaboration and help in my projects and their friendship. A special thanks goes to Rafael Bidese, his wife, Gabi Itokazu, their children, Filipe and Marcelo, and their extended family, for being a second family for me here in Auburn. I have too many amazing friends outside of Auburn to thank, but I would like to extend a special thank you to Chandler Compton, Courtney Eagle, Kirsta Moussad, Aidan Ryan, and Jennifer Robin Ryan.

Thank you to the research staff in the Department of Biosystems Engineering, including Bobby Bradford, James Johnson, Caroline Whiting, and the undergraduate staff.

Finally, I would like to thank my family for the support they have given me throughout my education, both undergraduate and graduate. My parents, Michael and Amy Cassity, for instilling in me value of education and hard work and for their financial support, my sister, Michael Kathryn Cassity-Allen, for always believing and telling me I was smart enough, my

grandad, Jim DePoyster, for helping me fund my undergraduate education, and my grandad, Floyed Cassity, for always being excited about and interested in my education.

Table of Contents

List of Tables

List of Figures

Chapter 1. Introduction

Research in artificial intelligence (AI) has seen dramatic innovations in the field in recent years. As foundational research in AI for computer vision flourishes, accessible tools for applied computer vision projects have become available, such as Facebook's Detectron2, OpenCV, and opensource libraries such as Hugging Face's Transformers library (Bradski, 2000; Wolf et al., 2020; Wu et al., 2019). The developers, contributors to, and maintainers of these tools have given an invaluable resource to researchers in all disciplines: the ability to apply AI to challenges of efficiency, accuracy, and safety in their fields.

Agricultural and horticultural research is no exception to the influence of applied AI research, encompassing applications of the closely related fields of computer vision, machine learning, and deep learning. High-throughput phenotyping has been of particular interest, with the push for phenotyping technology development to meet the pace of genotyping technology development (Ninomiya, 2022; White et al., 2012). This is reflected in the many current research centers today dedicated to plant phenotyping (Ninomiya, 2022). Distinct applications of computer vision, machine learning, and deep learning to above-ground plant phenotyping include: plant disease detection (Roy & Bhaduri, 2021; Singh et al., 2020), stress detection (Ghosal et al., 2018; Ramos-Giraldo et al., 2020), yield prediction (Liu et al., 2017; Pothen & Nuske, 2016), and morphology, including plant architecture and measurements such as leaf area and biomass (Azzari et al., 2013; Chaivivatrakul et al., 2014; Jiang et al., 2016; Niknejad et al., 2023; Xiang et al., 2019). Below ground phenotyping of root systems is represented widely in literature of applied image processing and deep learning for image segmentation (Flavel et al.,

2017; Gerth et al., 2021; Mairhofer et al., 2012; Mooney et al., 2012; Phalempin et al., 2021; Soltaninejad et al., 2020).

The hand phenotyping of morphological traits both above and below ground can be labor intensive and destructive if the measurement technique requires the removal of the plant from its in situ environment. Hence, tools such as PlantCV (Gehan et al., 2017), a Python library for two-dimensional (2D) analysis of plant morphology, and Phenomenal (Artzet et al., 2019), a Python library for three-dimensional (3D) analysis of plant morphology have been developed for applying computer vision to the challenge of rapid assessment of plant morphology. Furthermore, numerous studies reflect the application of AI to morphological plant phenotyping, including crops such as corn (Chaivivatrakul et al., 2014), sorghum (Xiang et al., 2019), cotton (Jiang et al., 2016), and pine trees (Niknejad et al., 2023).

This work aims to explore new solutions to challenges in rapid phenotyping of plant morphology, both above and below ground. The second chapter of this thesis (Objective 1) builds upon extensive research in the challenge of segmenting root systems from X-ray computed tomography (CT) images for below ground plant organ morphological phenotyping. Segmenting roots from X-ray CT images allows for the non-destructive phenotyping of in situ root systems over time. The third chapter of this thesis (Objective 2) investigates a less investigated challenge: the application of computer vision and machine learning to the morphological phenotyping of above ground *Humulus lupulus* (hop) vines. Methods for measuring vine length, leaf area, and biomass were developed and evaluated, allowing for efficient and non-destructive measurement of these traits over time.

Objective 1. Root segmentation of X-ray computed tomography images of container horticultural plants using Mask R-CNN and DBSCAN-based 3D point cloud clustering

The objective of this study is to develop and evaluate a fully automated approach to root segmentation from X-ray CT images that is computationally and resource efficient and does not require large amounts of training data. We propose using a pretrained Mask R-CNN (He et al., 2020) model for instance segmentation of roots from 2D X-ray CT images. Training a model on 2D images as well as implementing transfer learning with a pretrained model greatly reduces the amount of training data necessary, reducing the time and labor needed to annotate data to apply this method to other images containing roots of different species. Masks from predicting on sequential CT images can be concatenated to produce a volumetric segmentation of the whole root system. After segmenting roots from 2D images, we also incorporated 3D information in post-processing to filter noise introduced by the 2D instance segmentation density-based spatial clustering of applications with noise (DBSCAN) (Sander et al., 1998). DBSCAN is an unsupervised machine learning algorithm that is robust to noise and not vulnerable to clustering mistakes caused by lateral roots, branching, and negative geotropism roots. Furthermore, we propose a fully automatic parameter tuning approach for DBSCAN. Thus, we were able to utilize 3D information in our approach without incurring considerable labor burdens. Our methodology does not require user input and is fully automated for segmenting unseen inputs.

To the best of our knowledge, our contribution is four-fold. This work evaluates instance segmentation on 2D X-ray CT images for volumetric segmentation of roots in growing mediums. Second, we trained and evaluated one deep learning model on two different species. Third, we investigated training and predicting on 2D images along all axes of the soil column and

combining predictions from multiple axes. Fourth, we implemented DBSCAN with fully automated parameter tuning for automatic noise removal.

Objective 2. High-throughput phenotyping of *Humulus lupulus* in greenhouse using stereo vision, semantic segmentation, and 3D point cloud analysis

The objective of this study is to develop a semi-automated approach for measuring phenotypic traits of hops including vine length, leaf area, and biomass. We propose fine-tuning a 2D SegFormer (Xie et al., 2021) model from the Hugging Face Transformers library (Wolf et al., 2020) for semantic segmentation of hop plant from 2D images. Stereolabs' ZED 2 (Stereolabs, San Francisco, CA) camera was used to capture ZED SVO files. Left right stereo image pairs and corresponding point clouds were retrieved from the SVO files. Hop plants were segmented from the left stereo image using the fine-tuned SegFormer model and the 2D segmentations were applied to the 3D point clouds. A semi-automated pipeline was developed and assessed to measure vine length, leaf area, and biomass from the point clouds. The V-shaped architecture imposed on the hops by growing them on a trellis was leveraged for individual vine separation and length measurement using random sample consensus (RANSAC) (Fischler & Bolles, 1981). Furthermore, total leaf area for each plant was estimated by summing the total triangle areas in a triangle mesh of the segmented 3D point cloud. Biomass results regressed from voxel count and leaf area were compared.

To the best of our knowledge, this is the first work to employ computer vision and machine learning for automating individual plant segmentation and 3D reconstruction for morphological phenotypic trait extraction for hops.

References

Artzet, S., Chen, T.-W., Chopard, J., Brichet, N., Mielewczik, M., Cohen-Boulakia, S., Cabrera-Bosquet, L., Tardieu, F., Fournier, C., & Pradal, C. (2019). Phenomenal: An automatic open source library for 3D shoot architecture reconstruction and analysis for image-based plant phenotyping. *BioRxiv*. https://doi.org/10.1101/805739

Azzari, G., Goulden, M., & Rusu, R. (2013). Rapid Characterization of Vegetation Structure with a Microsoft Kinect Sensor. *Sensors*, *13*(2), 2384–2398. https://doi.org/10.3390/s130202384

Bradski, G. (2000). *The OpenCV Library*. Dr. Dobb's Journal of Software Tools.

Chaivivatrakul, S., Tang, L., Dailey, M. N., & Nakarmi, A. D. (2014). Automatic morphological trait characterization for corn plants via 3D holographic reconstruction. *Computers and Electronics in Agriculture*, *109*, 109–123. https://doi.org/10.1016/j.compag.2014.09.005

Fischler, M. A., & Bolles, R. C. (1981). Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, *24*(6), 381-395. https://doi.org/10.1145/358669.358692

Flavel, R. J., Guppy, C. N., Rabbi, S. M. R., & Young, I. M. (2017). An image processing and analysis tool for identifying and analysing complex plant root systems in 3D soil using non-destructive analysis: Root1. *PloS one*, *12*(5), e0176433. https://doi.org/10.1371/journal.pone.0176433

Gehan, M. A., Fahlgren, N., Abbasi, A., Berry, J. C., Callen, S. T., Chavez, L., Doust, A. N., Feldman, M. J., Gilbert, K. B., Hodge, J. G., Hoyer, J. S., Lin, A., Liu, S., Lizárraga, C., Lorence, A., Miller, M., Platon, E., Tessman, M., & Sax, T. (2017). PlantCV v2: Image analysis software for high-throughput plant phenotyping. *PeerJ*, *5*, e4088. https://doi.org/10.7717/peerj.4088

Gerth, S., Claußen, J., Eggert, A., Wörlein, N., Wininger, M., Wittenberg, T., & Uhlmann, N. (2021). Semiautomated 3D Root Segmentation and Evaluation Based on X-Ray CT Imagery. *Plant Phenomics*, *2021*. https://doi.org/10.34133/2021/8747930

Ghosal, S., Blystone, D., Singh, A. K., Ganapathysubramanian, B., Singh, A., & Sarkar, S. (2018). An explainable deep machine vision framework for plant stress phenotyping. *Proceedings of the National Academy of Sciences*, *115*(18), 4613–4618. https://doi.org/10.1073/pnas.1716999115

He, K., Gkioxari, G., Dollar, P., & Girshick, R. (2020). Mask R-CNN. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *42*(2), 386–397. https://doi.org/10.1109/TPAMI.2018.2844175

Jiang, Y., Li, C., & Paterson, A. H. (2016). High throughput phenotyping of cotton plant height using depth images under field conditions. *Computers and Electronics in Agriculture*, *130*, 57–68. https://doi.org/10.1016/j.compag.2016.09.017

Liu, S., Cossell, S., Tang, J., Dunn, G., & Whitty, M. (2017). A computer vision system for early stage grape yield estimation based on shoot detection. *Computers and Electronics in Agriculture*, *137*, 88–101. https://doi.org/10.1016/j.compag.2017.03.013

Mairhofer, S., Zappala, S., Tracy, S. R., Sturrock, C., Bennett, M., Mooney, S. J., & Pridmore, T. (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. https://doi.org/10.1104/pp.111.186221

Mooney, S. J., Pridmore, T. P., Helliwell, J., & Bennett, M. J. (2012). Developing X-ray Computed Tomography to non-invasively image 3-D root systems architecture in soil. *Plant and Soil*, *352*(1–2), 1–22. https://doi.org/10.1007/s11104-011-1039-9

Niknejad, N., Bidese-Puhl, R., Bao, Y., Payn, K. G., & Zheng, J. (2023). Phenotyping of architecture traits of loblolly pine trees using stereo machine vision and deep learning: Stem diameter, branch angle, and branch diameter. *Computers and Electronics in Agriculture*, *211*, 107999. https://doi.org/10.1016/j.compag.2023.107999

Ninomiya, S. (2022). High-throughput field crop phenotyping: current status and challenges. *Breeding Science*, *72*(1), 3-18. https://doi.org/10.1270/jsbbs.21069

Phalempin, M., Lippold, E., Vetterlein, D., & Schlüter, S. (2021). An improved method for the segmentation of roots from X-ray computed tomography 3D images: Rootine v.2. *Plant Methods*, *17*, 39. https://doi.org/10.1186/s13007-021-00735-4

Pothen, Z. S., & Nuske, S. (2016). Texture-based fruit detection via images using the smooth patterns on the fruit. *2016 IEEE International Conference on Robotics and Automation (ICRA)*, 5171–5176. https://doi.org/10.1109/ICRA.2016.7487722

Ramos-Giraldo, P., Reberg-Horton, C., Locke, A. M., Mirsky, S., & Lobaton, E. (2020). Drought stress detection using low-cost computer vision systems and machine

learning techniques. *IT Professional*, *22*(3), 27–29. https://doi.org/10.1109/MITP.2020.2986103

Roy, A. M., & Bhaduri, J. (2021). A deep learning enabled multi-class plant disease detection model based on computer vision. *AI*, *2*(3), 413–428. https://doi.org/10.3390/ai2030026

Sander, J., Ester, M., Kriegel, H.-P., & Xu, X. (1998). Density-based clustering in spatial databases: The algorithm gdbscan and its applications. *Data Mining and Knowledge Discovery*, *2*(2), 169–194. https://doi.org/10.1023/A:1009745219419

Singh, V., Sharma, N., & Singh, S. (2020). A review of imaging techniques for plant disease detection. *Artificial Intelligence in Agriculture*, *4*, 229–242. https://doi.org/10.1016/j.aiia.2020.10.002

Soltaninejad, M., Sturrock, C. J., Griffiths, M., Pridmore, T. P., & Pound, M. P. (2020). Three dimensional root CT segmentation using multi-resolution encoder-decoder networks. *IEEE Transactions on Image Processing*, *29*, 6667–6679. https://doi.org/10.1109/TIP.2020.2992893

White, J. W., Andrade-Sanchez, P., Gore, M. A., Bronson, K. F., Coffelt, T. A., Conley, M. M., Feldmann, K. A., French, A. N., Heun, J. T., Hunsaker, D. J., Jenks, M. A., Kimball, B. A., Roth, R. L., Strand, R. J., Thorp, K. R., Wall, G. W., & Wang, G. (2012). Field-based phenomics for plant genetics research. *Field Crops Research*, *133*, 101–112. https://doi.org/10.1016/j.fcr.2012.04.003

Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., Cistac, P., Rault, T., Louf, R., Funtowicz, M., Davison, J., Shleifer, S., von Platen, P., Ma, C., Jernite, Y., Plu, J., Xu,

C., Le Scao, T., Gugger, S., … Rush, A. (2020). Transformers: State-of-the-Art Natural Language Processing. *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, 38–45. https://doi.org/10.18653/v1/2020.emnlp-demos.6

Wu, Y., Kirillov, A., Massa, F., Lo, W.-Y., & Girshick, R. (2019). *Detectron2*. https://github.com/facebookresearch/detectron2

Xiang, L., Bao, Y., Tang, L., Ortiz, D., & Salas-Fernandez, M. G. (2019). Automated morphological traits extraction for sorghum plants via 3D point cloud data analysis. *Computers and Electronics in Agriculture*, *162*, 951–961. https://doi.org/10.1016/j.compag.2019.05.043

Xie, E., Wang, W., Yu, Z., Anandkumar, A., Alvarez, J. M., & Luo, P. (2021). SegFormer: Simple and efficient design for semantic segmentation with transformers. *Advances in Neural Information Processing Systems*, *34*, 12077-12090. https://doi.org/10.48550/arXiv.2105.15203

Chapter 2. Root segmentation of X-ray computed tomography images of container horticultural plants using Mask R-CNN and DBSCAN-based 3D point cloud clustering

2.1 Abstract

X-ray computed tomography (CT) is a powerful tool for in situ plant root system architecture (RSA) characterization. Accurate root segmentation from CT images is integral to studying RSA. Research studies on segmenting roots from CT images have been mainly limited to image processing-based approaches which may require parameter tuning and often lack common segmentation metrics, e.g., dice and IoU. A recent deep learning approach utilizes a volumetric encoder-decoder network to achieve a high dice score and IoU. However, training a volumetric encoder-decoder model is dependent on the availability of fully annotated scans of the soil column, obtaining which can be time-consuming, tedious, and resource-intensive. In this study, an efficient method using deep learning-based instance segmentation in conjunction with density-based spatial clustering of applications with noise (DBSCAN)-based filtering was developed and evaluated for container horticultural plants. A two-dimensional Mask R-CNN model was fine-tuned on CT images to detect root instances. Sequential two-dimensional predictions were combined to produce a volumetric segmentation of the root system. Merged predictions along multiple axes of the soil column were compared to predictions made along one axis of the soil column. DBSCAN was used to filter noise from the volumetric segmentation with an automated parameter tuning technique. The proposed method was evaluated on scans of

poinsettias and onions and achieved average scores of 0.734, 0.868, 0.749, and 0.669 for precision, recall, dice, and IoU, respectively, utilizing only 1% of the training dataset. The proposed method requires significantly fewer annotations compared to the volumetric encoder-decoder approach, making it more accessible to use.

2.2 Introduction

Crop phenotyping is important for breeding resource-efficient crops, but conventional phenotyping methods are often limited to what can be seen above ground. However, root systems play a vital role in many plant functions, and a deeper understanding of how root systems interact with the environment can lead to the development of more resilient crops. Roots respond to both biotic and abiotic stress, reacting by changing their architecture in response to various conditions (Downie et al., 2015; Smith & De Smet, 2012). Moreover, root system architecture (RSA) determines a plant's distribution of resource capture and transport functions (Koevoets et al., 2016). Therefore, quantitatively capturing RSA is an important part of plant phenotyping. Invasive approaches such as removing the root system from the soil have been used to phenotype RSA. However, these methods deform the root's three-dimensional (3D) structure and often cause a loss of root mass either through loss of fine roots during removal or after removal from soil (Livesley, 1998; van Noordwijk & Floris, 1979). Other studies have indicated error in overestimating biomass measurements due to the adherence of non-root particles (Janzen, 2002). Furthermore, invasive methods do not allow for replanting the roots in the same 3D configuration they occupied before, making it impossible to analyze the root system's spatial growth over time (Mooney et al., 2012). Research in segmentation of roots grown in soil or a

growing medium (Kang et al., 2021; Shen et al., 2020; Wang et al., 2019; Yu et al., 2023) or a gel medium (Gaggion et al., 2021; Thesma & Mohammadpour Velni, 2022) from two-dimensional (2D) images taken of cross sections of soil profiles is common. These methods offer a solution for low cost and rapid imaging of in situ root systems over time, however, they do not capture a 3D view of the root system.

X-ray computed tomography (CT), originally developed by G.N. Hounsfield for medical and clinical imaging (Mooney et al., 2012), offers a tool for non-invasive RSA phenotyping. Advances in CT technology that allow for higher resolution imaging have given X-ray CT a place in phenotyping RSA (Mooney et al., 2012). Furthermore, X-ray CT has been evidenced as a tool to detect both biotic and abiotic stress (Tollner, 1991). A common method for segmenting root systems from X-ray CT images is thresholding (Galkovskyi et al., 2012; Shao et al., 2021). However, characteristics of root systems and the substrate they grow in pose significant challenges to accurate segmentation of root systems from X-ray CT images, including, overlap in attenuation values between root matter and water or other organic matter in the substrate and root topology such as negative geotropism roots (Mooney et al., 2012). It is important to note that using X-ray CT is appropriate for plants grown in containers that can be placed in the X-ray scanner CT without disturbing the roots and the soil or growing medium.

Many 2D and 3D image-processing based segmentation methods include methodologies to mitigate the inaccuracies introduced by overlapping attenuation values and diverse root topology such as RooTrak (Mairhofer et al., 2012), Root1 (Flavel et al., 2017), Rootine v.2 (Phalempin et al., 2021), and RootForce (Gerth et al., 2021). Note, when used in this work "accuracy" refers to multiple computer vision metrics, including precision, recall, dice, and IoU

as opposed to the number of correct predictions divided by the total number of predictions. Please refer to the Evaluation section for the definitions of precision, recall, dice, and IoU.

RooTrak employs a top-down tracking approach across sequential CT images to mitigate error incurred by overlapping attenuation values as well as varying attenuation values of root material within the root system. By using the level-set method (also employed in a work by Tabb et al. in 2018), local models of the distribution of the attenuation values are made for each image. The level-set function (Sethian, 1999) is updated from each previous image to be applied to the next image in the stack using the Jensen-Shannon (JS) divergence (Lin, 1991). Because the local model for each image is affected by the previous, error incurred can accumulate as RooTrak traverses down the image stack. To remedy this, the model is only updated if the root shapes of the current image are similar to the root shapes in the previous image (Mooney et al., 2012). To use RooTrak, a user first initializes the tracking by clicking in the root segmentation in the first image of the image stack.

Root1 uses ImageJ for 3D segmentation of roots. First, the CT images are normalized and then enlarged to avoid creating discontinuity of the root system during a later erosion process. Next, ImageJ's "Find Edges" tool is used to ameliorate the partial volume effect (PVE), caused by the occurrence of both root and other matter being contained in one voxel in the CT image. After the input image has been prepared, ImageJ's "Median (3D)" is employed to remove small objects attached to the root system before a bilevel thresholding is implemented. After the segmentation by thresholding is complete, the ImageJ plugin "Erode (3D)" can be used to remove continuity between the segmented roots and poor space. Finally, the segmentation is

exported and VGStudioMax is used to select the root system based on connectivity with a user defined seed point.

Rootine v.2 employs the ImageJ's "Tubeness filter" and the ImageJ plugins "MorpholibJ" and "3D Hysteresis Thresholding" and the optional plugin "Attenuation correction". In addition, the "3D Non-local Means (NLM)" filter from ITK is used. Rootine v.2 offers improvement upon Rootine v.1 (Gao et al., 2019) including pot wall detection and removal in data pre-processing and background removal by masking out voxels with a significantly different gray value than the average gray value of the segmented material. Rootine v.2 further improves upon Rootine v.1 by the introduction of a new method to calculate the lower threshold used during hysteresis thresholding in root segmentation as well as by implementing a post-processing step to identify false negatives. Segmentations that are not continuously connected to the root system are evaluated by their vesselness (shape) and size. Using vessleness as filter during post-processing was first proposed by Frangi et al. in 1998. False positives are discarded from the segmentation while false negatives are included in the segmentation. Therefore, Rootine v.2 avoids discarding portions of the root system because they are not connected.

Vesselness is also employed in RootForce, for which Rootine v.1 was a precursor. First, the pot wall is removed by thresholding by the pot wall thickness and attenuation value. Then, filtering by vesselness is employed for smaller root structures. However, the authors noticed that the original method defined by Frangi et al. was not effective for larger root structures such as storage roots. These larger roots are segmented by their 3D variance, implemented by a 3D-Gaussian filter. Therefore, RootForce is designed to segment both fine and coarse roots.

The above-described methods all have one thing in common: the features used for segmentation of roots in soil or growing medium are hand selected. Therefore, methods that are successful for one species of plant and type of soil or growing medium may not be successful for other species of plants and soil or growing medium. In contrast, deep learning algorithms can extract many features of root systems, including, size, shape, and texture, automating the segmentation process, and requiring less human intervention. In other words, the deep learning model selects features (possibly undetectable by humans) by quantitative optimization to achieve optimal segmentation accuracy. Furthermore, deep learning models can be trained on diverse datasets, allowing the model to learn complex features of roots for more flexibility in use.

Soltaninejad et al. (2020) developed a deep learning-based parallel pipeline that combined both high- and low-resolution encoder-decoders with a multi-loss training approach for volumetric segmentation. This was the first time a deep learning method was applied to CT images for root segmentation. This method was evaluated against other methods of deep learning-based approaches to semantic segmentation and Root1 and proved to result in the best scores for dice and IoU (0.740 and 0.588, respectively) using the multi-loss multi-resolution model. Though other image processing-based and deep learning-based approaches scored higher for precision and recall, scoring high for one of these metrics does not indicate an accurate segmentation method. For example, a method resulting in a high precision score but with a low score for recall would indicate the root system was under-segmented. Likewise, a high recall score with a low precision score would indicate the root system was segmented from the soil, but with low precision and many false positives (noise). High dice and IoU scores indicate a balanced approach that does not over or under-segment and scores comparatively for precision

and recall. Therefore, the multi-loss multi-resolution encoder-decoder network provided a fully automatic approach that avoided both over and under-segmentation with a precision and recall of 0.733 and 0.750, respectively.

One drawback to volumetric segmentation methods is the need for large amounts of labeled training data. Not only does the user have to provide annotated ground truth data for evaluating the accuracy of the segmentation, but fully annotated scans must also be provided to train a volumetric model. Annotating large amounts of data can be time-consuming, tedious, and resource intensive, and introduces bias into the training data with human error. Alternatively, when training a model on 2D images, the entire CT scan does not have to be annotated to train the model, only for model evaluation. Therefore, a subset of 2D images can be selected from the 3D scan for training, greatly reducing the amount of annotation work and possible errors in the ground truth annotations used for training.

The objective of this study is to develop and evaluate a fully automated approach to root segmentation from X-ray CT images that is computationally and resource efficient and does not require large amounts of training data. We propose using a pretrained Mask R-CNN (He et al., 2020) model for instance segmentation of roots from 2D X-ray CT images. Training a model on 2D images as well as implementing transfer learning with a pretrained model greatly reduces the amount of training data necessary, reducing the time and labor needed to annotate data to apply this method to other images containing roots of different species. Masks from predicting on sequential CT images can be concatenated to produce a volumetric segmentation of the whole root system. After segmenting roots from 2D images, we also incorporated 3D information in post-processing to filter noise introduced by the 2D instance segmentation density-based spatial

clustering of applications with noise (DBSCAN) (Sander et al., 1998). DBSCAN is an unsupervised machine learning algorithm that is robust to noise and not vulnerable to clustering mistakes caused by lateral roots, branching, and negative geotropism roots. Furthermore, we propose a fully automatic parameter tuning approach for DBSCAN. Thus, we were able to utilize 3D information in our approach without incurring considerable labor burdens. Our methodology does not require user input and is fully automated for segmenting unseen inputs.

To the best of our knowledge, our contribution is four-fold. This work evaluates instance segmentation on 2D X-ray CT images for volumetric segmentation of roots in growing mediums. Second, we trained and evaluated one deep learning model on two different species. Third, we investigated training and predicting on 2D images along all axes of the soil column and combining predictions from multiple axes. Fourth, we implemented DBSCAN with fully automated parameter tuning for automatic noise removal.

2.3 Materials and Methods

2.3.1 Data Collection

All plant root samples used in this study were scanned using a high-resolution X-ray CT scanner (XTH 225 ST, Nikon, Melville, NY) housed in the Shared Material and Instruments Facility at Duke University (Durham, NC). The instrument utilized a tungsten target, white beam, and conical beam with a detection panel containing 2000 × 2000 pixels. A Feldkamp cone-based CT algorithm was applied to convert the X-ray radiographs into 2D reconstructed slices. A total of 2,200 radiographs were collected from each scan by rotating the sample by 0.164 degrees (2 ×

frame averaging to reduce image noise). X-ray settings were adjusted to improve the image quality when scanning plant samples of differing volumes and moisture contents. Poinsettia (*Euphorbia pulcherrima* "Majestic Red") was asexually propagated and rooted in an engineered foam medium (87-50010 RootCube Wedge; Oasis Grower Solutions, Kent, OH, USA). The poinsettia were scanned at 80 kV, 140 µA. The two data sets selected from the poinsettia scans for further processing will be referred to herein as "Poinsettia1" and "Poinsettia2". Onion (*Allium cepa*) was grown in either a polyacrylic tube (6.35 cm internal diameter) or a custom peat-based sleeve. One onion sample was scanned at 95 kV, 120 mA. The second onion sample was scanned at a higher moisture content at 110 kV, 120 mA. The two datasets selected from the onion scans for further processing will be referred to herein as "Onion1" and "Onion2". The images contain a voxel resolution of 30 µm and 55 µm for the poinsettia and onion, respectively.

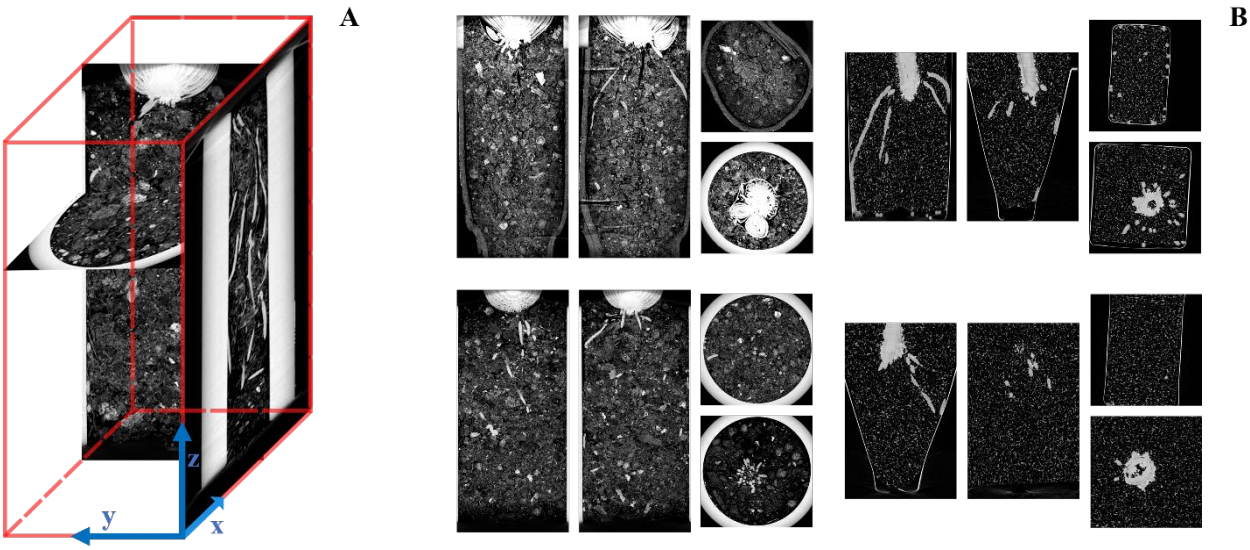


Figure 2.1 (A) Sample images and their orientations from the Onion2 soil column. Images are from slices along the x, y, and z axes. (B) Sample images from the Onion1 (top left), Poinsettia1 (top right), Onion2 (bottom left), and Poinsettia2 (bottom right) soil columns. For each image set, a sample image from the x axis (left), the y axis (middle), and two images from the z axis (right) of the soil column are displayed.

2.3.2 Data Pre-Processing

Two datasets were made to test whether training a Mask R-CNN model on images from all axes of the soil column and combining prediction results from multiple axes of the soil column improved segmentation results. The first dataset was produced from the original X-ray CT images. These images were taken along the z axis of the soil column. Ground truth annotations were obtained by manual annotation of root instances using COCO Annotator (Brooks, 2019). The images and corresponding ground truth masks were resized to 920 × 920 pixels while preserving the aspect ratio of each image. The second dataset was produced by transforming both the input images and annotation masks from the first dataset. The images and masks were stacked sequentially to produce a volumetric image and mask for each plant. Next,

these images were sliced along the x, y, and z axes of the volume to produce images and corresponding masks from each view of the soil column. Figure 1 illustrates images from the x, y, and z axes and their orientation in the volumetric image and displays sample images from each plant.

In addition to producing images from all axes of the soil column, data augmentation was performed by random cropping and rotation. Four 512 × 512 pixel tiles were randomly cropped from each input image and its corresponding ground truth mask and rotated randomly by 0°, 15°, 30°, 45°, 60°, 90°, 180°, or 270°. The instances of roots in each mask were found using *findContours* function from OpenCV 4.7.0 (Bradski, 2000) and the segmentations for each instance were saved in JSON files in the COCO format for training. Figure 2.2 illustrates the pipeline used to prepare the datasets.

To perform cross validation, th datasets were split into two groups: Dataset1 (Onion1 and Poinsettia1) and Dataset2 (Onion2 and Poinsettia2). Therefore, four total datasets were used: Dataset1 z, Dataset1 xyz, Datset2 z, and Dataset2 xyz. Note, z refers to datasets containing tiles from the z view and xyz refers to datasets containing tiles from the x, y, and z views. The Mask R-CNN models trained on Onion1 and Poinsettia1 were used for prediction on the images from Onion2 and Poinsettia2 and vice versa.

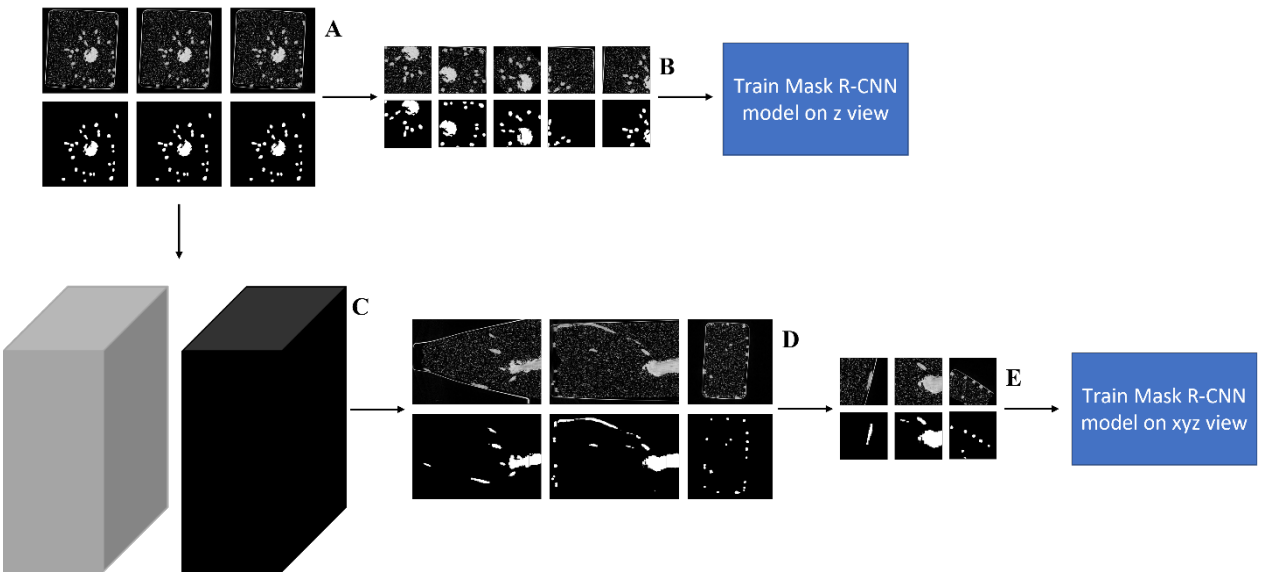


Figure 2.2 Data preparation pipeline including: (A) original X-ray CT input images and ground truth binary masks, (B) tiled and rotated input images and ground truth binary masks for the z view, (C) volumetric input images and ground truth binary masks, (D) input images and ground truth binary masks for the x, y, and z views of the soil column, (E) tiled and rotated input images and ground truth binary masks for the x, y, and z views.

2.3.3 2D instance segmentation with Mask R-CNN

Mask R-CNN models were trained using Facebook AI Research's Detectron2 (Wu et al., 2019) and Python (van Rossum, 2022) 3.10.9. The backbone and pretrained COCO Instance Segmentation model mask_rcnn_50_FPN_1x was used for training with a learning rate of 0.001, batch size of 64, and maximum iterations of 1000. The hyperparameters were selected by grid search. The default optimizer stochastic gradient descent (SGD) was used.

After the Mask R-CNN models were trained, they were used to predict root instances in each scan. During prediction, each image was tiled into 512 × 512 pixel images with a side overlap of 100 pixels. The predicted masks for each tile were recombined using a logical OR

operation to make one predicted mask for each image. The models trained on images from the z view were used to make instance predictions on images from the z view. The models trained on images from xyz view were used to make instance predictions on images from all three views.

Sequential predicted masks were concatenated to produce a volumetric mask for the entire soil column. Seven different volumetric masks were made by combining the predictions on images from all three views, including: an x prediction, y prediction, z prediction, xy prediction, xz prediction, yz prediction, and xyz prediction. In addition, there were volumetric masks created by combining predictions made on the z view by the model trained on the z view. Therefore, eight total predictions existed for each plant.

To compare our method against a baseline approach, the soil columns were thresholded using a global threshold. A nested for loop was used to test each possible threshold from 0 to 255 with a step size of 5.

2.3.4 3D point cloud clustering and noise removal with DBSCAN

The proposed method aimed to incorporate 3D information in its methodology without needing to train a model using 3D images or relying on top-down or bottom-up connectivity. Therefore, Euclidean-based clustering for noise removal was explored and a filtering method using DBSCAN was implemented. Unlike most clustering methods, DBSCAN clusters data points based on density instead of distance, meaning the algorithm clusters continuous volumes of homogenous density in space similarly to how a human would. DBSCAN also accounts for continuity without using a top-down or bottom-up approach. Furthermore, DBSCAN is an unsupervised machine learning algorithm that can be implemented independent of the amount of training data used to

train our instance segmentation model. Although DBSCAN is commonly used for noise removal, it can be difficult to select parameters to optimize noise removal for DBSCAN. Therefore, methodology was developed to automatically compute the input parameters for the DBSCAN algorithm and remove noise from the segmentation.

Before clustering our segmentation with DBSCAN, the dataset was downsampled to increase efficiency. Three dimensional NumPy arrays consisting of a volumetric binary mask were made from the prediction segmentations saved in the JSON files. First, the segmentation contours were converted into a point cloud using Open3D 0.16.0 (Zhou et al., 2018). Open3D was used to downsample the data with voxel downsampling. A voxel size of 1.5 was used.

DBSCAN requires two input parameters, *minPts* (the minimum number of neighbors a point must have to not be considered noise) and ε (the radius in which DBSCAN can search for neighbors). Sander et al. (1998) offered heuristics for choosing both parameters. It is common to use the *minPts* = 2 × dimensions. For clustering in 3D dimensional space, *minPts* would equal six. This was experimentally found to be the optimal choice for *minPts* for the dataset through a grid search for *minPts*. Epsilon (ε) can be chosen by plotting the sorted k-nearest-neighbor distances computed for each point in the dataset (Schubert et al., 2017). This plot will have an "elbow" or "knee" point which indicates a value for ε. This was computed using the Python package *kneed* after smoothing the k-nearest-neighbor distances plot with a moving average filter. The Python package was an implementation of the kneedle algorithm (Satopaa et al., 2011). Any data point without the required number of minimum points in the radius ε or not in the radius of a core point would be removed as noise.

Despite using the above methods for automatic parameter tuning, there was still noise present in the scans. The DBSCAN algorithm only removed noise if it belonged to a cluster consisting of six or fewer data points (voxels segmented as root). The predictions obtained through instance segmentation with Mask R-CNN contained noise belonging to clusters that consisted of more than six data points. Therefore, the technique described above to select ε was adapted to filter possible noise clusters that were not removed by DBSCAN. The sorted number of points in each cluster were plotted and the knee point was computed using *kneed* after smoothing the plot with a moving average filter. This knee point indicated the point of maximum curvature on the plot of number of points in each cluster. It was assumed that noise clusters would contain significantly fewer data points than non-noise clusters, creating a clear point of maximum curvature. Clusters that were larger than the cluster size at the point of maximum curvature were kept while clusters that were smaller than the cluster size at the point of maximum curvature were removed as noise.

After noise was filtered from the segmentation, the point cloud was upsampled to compute the segmentation accuracy metrics. First, the point cloud was converted back into a 3D NumPy array consisting of a volumetric binary mask. Then, dilation and erosion with a five-by-five kernel and one iteration was performed using OpenCV for the 2D masks along the first axis. Dilation and erosion operations were then performed for each 2D mask along the third axis. This upsampled the segmentation, and it could then be used to compute the accuracy of the method. Examples of downsampled and upsampled 2D masks can be seen in Figure 2.3.

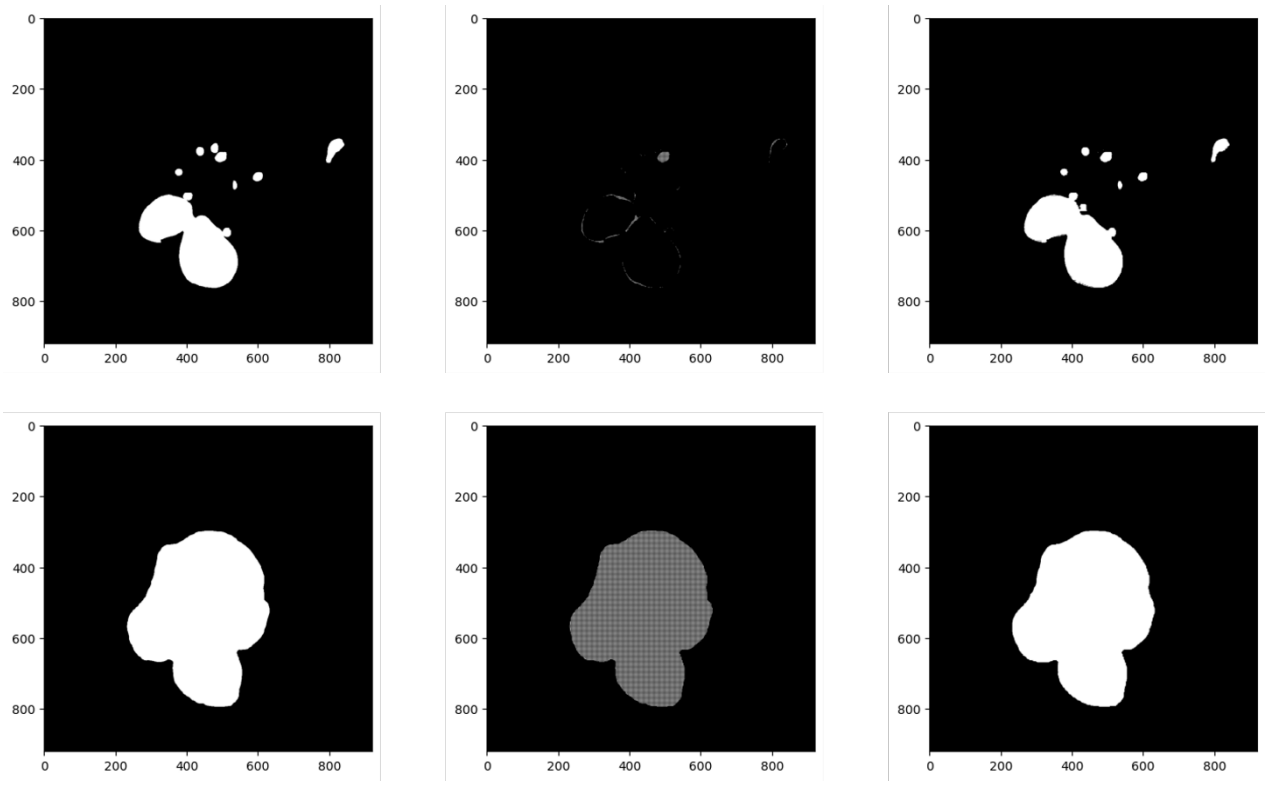


Figure 2.3 Sample binary masks from Onion1 predictions: original predictions (left), down sampled predictions (middle), predictions restored by dilation and erosion (right). Note, the middle images may appear to be grayscale due to aliasing.

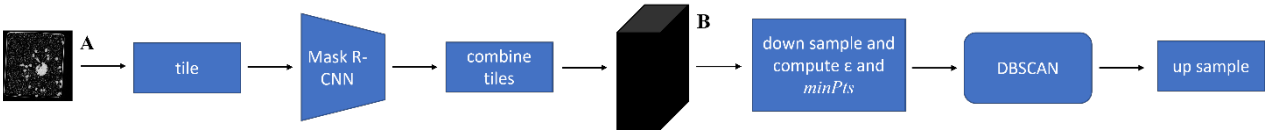


Figure 2.4 Segmentation method for the z view including: (A) input images from the z view, (B) segmented volume from concatenating sequential z predictions.

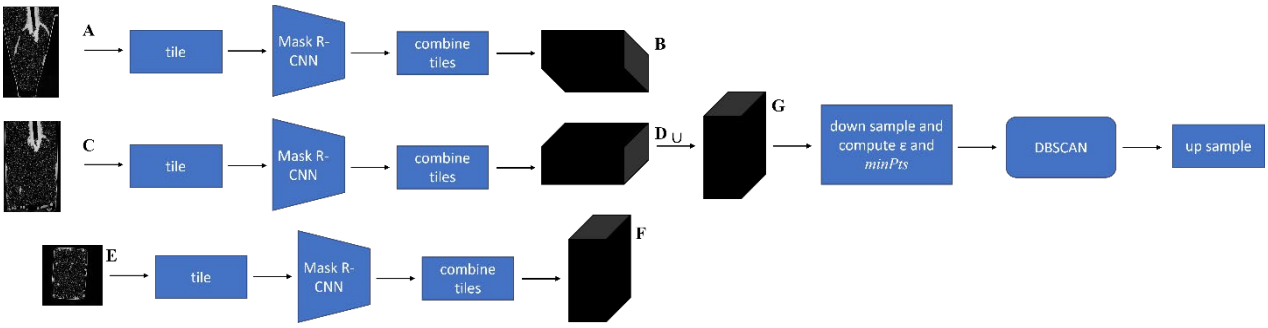


Figure 2.5 Segmentation method for the xyz view including: (A) input images from the x view, (B) segmented volume from concatenating sequential x predictions, (C) input images from the y view, (D) segmented volume from concatenating sequential y predictions, (E) input images from the z view, (F) segmented volume from concatenating sequential z predictions, (G) segmented volume by combining multiple views.

2.3.5 Evaluation

Precision, recall, dice (equivalent to F_1), and IoU were used to evaluate the proposed segmentation and filtering method. These are commonly used metrics that describe both the successes (true positives and true negatives) and failures (false positives and false negatives) of the segmentation as well as overall segmentation accuracy. The definitions of these metrics are:

$$Precision = \frac{TP}{TP + FP} \quad (1)$$

$$Recall = \frac{TP}{TP + FN} \quad (2)$$

$$Dice = \frac{2|G \cap S|}{|G| + |S|} \quad (3)$$

$$IoU = \frac{|G \cap S|}{|G \cup S|} \quad (4)$$

where

TP = Number of correctly identified positive voxels (true positives),

FP = Number of incorrectly identified positive voxels (false positives),

FN = Number of incorrectly identified negative voxels (false negatives),

G = Ground truth segmentation, and

S = Predicted segmentation.

These metrics were computed for both complete scans and subsets of the entire scans. The subsets were selected to only include root material and represented the entire width and length of the original scans, but with truncated height (z axis) to exclude non-root plant material at the top of the scans. These truncated scans were evaluated because the presence of non-root plant material in the complete scans would result in a large number of true positives, causing an inflated score for precision, and therefore, an inflated score for both dice and IoU. Scores closer to 1 indicate better segmentation accuracy for all metrics. The metrics were computed by iterating through each 1×1×1 voxel in the volumetric ground truth binary mask and the corresponding prediction binary mask.

2.3.6 Environment

The experiments were conducted on a laptop (Creator M16, MSI, Taiwan) with an Intel Core i7 CPU at 2.30 GHz, 32 GB of RAM, and an NVIDIA GeForce RTX 3050 GPU. Python (van Rossum, 2022) 3.10.9 was used for the experiments.

2.4 Results

The following results were obtained for evaluating instance segmentation with a fine-tuned Mask R-CNN model for volumetric segmentation of roots in soil. Table 2.1 presents results for the entire scans while Table 2.2 presents results for truncated scans. Results from models trained on the z and xyz view and predictions from the z view are featured. These training and prediction views achieved the highest scores for precision, recall, dice, and IoU of the eight total predictions described in the Materials and Methods. Table 2.2 also includes results from models trained on the xyz view and predictions from the combined x, y, and z views due to its high score for recall. Predicting on all three views increases the chance of identifying root pixels, as it provides three opportunities (the maximum amount) for the model to make predictions for the same voxel in the soil column. The best result for each accuracy metric is bolded. For complete tables including results from all possible prediction view combinations, please refer to Table S1 and Table S2 in the Supplementary Materials.

Table 2.1 Highlighted segmentation accuracy metrics obtained by two-dimensional instance segmentation for the complete scans.

Training View(s)	Prediction View(s)	Plant	Precision	Recall	Dice	IoU
Z	Z	Poinsettia1	0.872	0.921	0.896	0.812
Z	Z	Onion1	0.910	0.926	0.918	0.848
Z	Z	Poinsettia2	0.900	0.976	0.937	0.881
Z	Z	Onion2	0.817	0.982	0.892	0.805
Average			**0.875**	**0.951**	**0.911**	**0.836**
XYZ	Z	Poinsettia1	0.882	0.895	0.888	0.799
XYZ	Z	Onion1	0.508	0.600	0.550	0.380
XYZ	Z	Poinsettia2	0.919	0.969	0.943	0.893
XYZ	Z	Onion2	0.853	0.957	0.902	0.822
Average			0.791	0.855	0.821	0.723

Table 2.2 Highlighted segmentation accuracy metrics obtained by two-dimensional instance segmentation for the truncated scans.

Training View(s)	Prediction View(s)	Plant	Precision	Recall	Dice	IoU
Z	Z	Poinsettia1	0.821	0.866	0.843	0.729
Z	Z	Onion1	0.533	0.877	0.663	0.496
Z	Z	Poinsettia2	0.808	0.956	0.876	0.779
Z	Z	Onion2	0.599	0.968	0.740	0.587
Average			0.690	0.917	0.781	0.648
XYZ	Z	Poinsettia1	0.806	0.833	0.819	0.694
XYZ	Z	Onion1	0.778	0.827	0.802	0.669
XYZ	Z	Poinsettia2	0.857	0.941	0.897	0.814
XYZ	Z	Onion2	0.674	0.947	0.787	0.649
Average			**0.779**	0.887	**0.826**	**0.706**
XYZ	XYZ	Poinsettia1	0.716	0.888	0.793	0.657
XYZ	XYZ	Onion1	0.613	0.904	0.731	0.576
XYZ	XYZ	Poinsettia2	0.713	0.973	0.823	0.699
XYZ	XYZ	Onion2	0.512	0.977	0.672	0.506
Average			0.639	**0.936**	0.755	0.610

Table 2.3 and Table 2.4 present results for filtering the predicted scans with DBSCAN-based 3D point cloud clustering. For complete tables including results from all possible prediction view combinations, please refer to Table S3 and Table S4 in the Supplementary

Materials. As seen in Table 2.1 and Table 2.2, training on the z and xyz view and predicting on the z view achieves the highest accuracy scores.

Table 2.3 Highlighted segmentation accuracy metrics obtained by two-dimensional instance segmentation and noise removal with DBSCAN for the complete scans.

Training View(s)	Prediction View(s)	Plant	Precision	Recall	Dice	IoU
Z	Z	Poinsettia1	0.870	0.912	0.891	0.803
Z	Z	Onion1	0.952	0.919	0.935	0.879
Z	Z	Poinsettia2	0.913	0.975	0.943	0.892
Z	Z	Onion2	0.891	0.972	0.929	0.868
Average			**0.906**	**0.945**	**0.925**	**0.860**
XYZ	Z	Poinsettia1	0.875	0.887	0.881	0.787
XYZ	Z	Onion1	0.467	0.590	0.521	0.353
XYZ	Z	Poinsettia2	0.925	0.968	0.946	0.898
XYZ	Z	Onion2	0.927	0.934	0.931	0.870
Average			0.798	0.845	0.820	0.727

Table 2.4 Highlighted segmentation accuracy metrics obtained by two-dimensional instance segmentation and noise removal with DBSCAN for the truncated scans.

Training View(s)	Prediction View(s)	Plant	Precision	Recall	Dice	IoU
Z	Z	Poinsettia1	0.813	0.835	0.824	0.700
Z	Z	Onion1	0.559	0.799	0.658	0.490
Z	Z	Poinsettia2	0.849	0.950	0.897	0.812
Z	Z	Onion2	0.723	0.945	0.819	0.694
Average			0.736	0.882	0.799	0.674
XYZ	Z	Poinsettia1	0.787	0.804	0.796	0.661
XYZ	Z	Onion1	0.838	0.743	0.788	0.649
XYZ	Z	Poinsettia2	0.881	0.927	0.904	0.824
XYZ	Z	Onion2	0.816	0.883	0.848	0.736
Average			**0.831**	**0.839**	**0.834**	**0.718**

Table 2.5 shows the dice scores and standard deviation for all four truncated plants when training with different percentages of training data. The results in Table 2.5 are from predictions on the truncated scans and include noise removal with DBSCAN. Predictions from the z view made from models trained on the z view had the highest mean dice score and the lowest standard deviation across different amounts of training data. Predictions from the xyz view made from models trained on the xyz view had the lowest mean dice score across different amounts of training data and the highest standard deviation. Figure 2.6 is a scatter plot of the mean dice scores for predicting on the z view with models trained on the z (Single) and xyz (Three) view, highlighting the variability in dice score across training dataset size for training on the xyz view.

Because training and predicting on the z view achieved the highest mean dice score, this view will be featured for further results. For the z view, 1% of the total dataset equaled 34 CT images for Dataset1 (Onion1 and Poinsettia1) and 35 CT images for Dataset2 (Onion2 and Poinsettia2).

Table 2.5 Mean dice scores of all four plants for training, prediction view(s) across percents of training data.

Percent of training data	**Z, Z**	**XYZ, X**	**XYZ, Y**	**XYZ, Z**	**XYZ, XY**	**XYZ, XZ**	**XYZ, YZ**	**XYZ, XYZ**
1	0.794	0.707	0.657	0.794	0.620	0.691	0.642	0.604
5	0.820	0.780	0.737	0.808	0.708	0.752	0.725	0.694
10	0.819	0.806	0.774	0.818	0.762	0.779	0.772	0.744
15	0.792	0.787	0.704	0.813	0.679	0.762	0.695	0.666
20	0.806	0.683	0.682	0.703	0.621	0.638	0.627	0.600
25	0.803	0.768	0.755	0.814	0.714	0.742	0.739	0.698
50	0.802	0.615	0.658	0.741	0.573	0.596	0.645	0.559
75	0.806	0.759	0.671	0.831	0.641	0.750	0.676	0.635
100	0.799	0.770	0.754	0.834	0.784	0.808	0.802	0.776
Mean ± SD	**0.805 ± 0.009**	0.742 ± 0.058	0.710 ± 0.043	0.795 ± 0.042	0.678 ± 0.066	0.724 ± 0.065	0.703 ± 0.058	0.664 ± 0.067

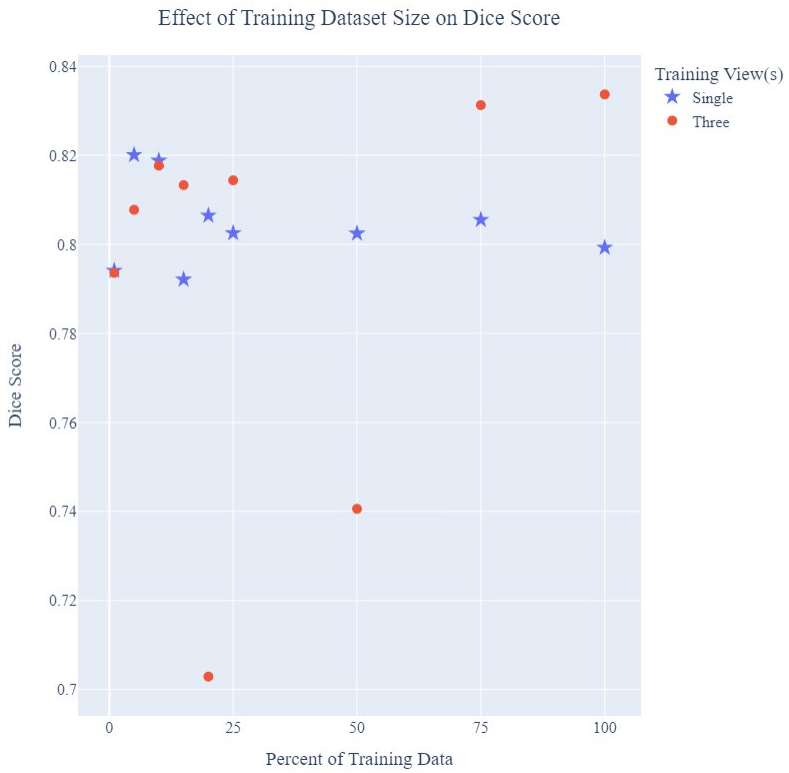


Figure 2.6 Mean dice scores of all four plants predictions made on the z view with models trained on the z and xyz view(s) across percents of training data.

Table 2.6 and Table 2.7 display the percent change before and after filtering noise with DBSCAN. It is important to note that not all plants dice and IoU scores improved after noise removal with DBSCAN. For both the truncated and complete scans, filtering noise with DBSCAN did not improve the overall segmentation accuracy (dice and IoU) for Onion1. However, the average across dice and IoU scores across all four plants improved. The recall score was lowered across all four scans. Figure 2.7 and Figure 2.8 emphasize the effect of non-root plant material when computing the segmentation accuracy metrics. The truncated scans experienced larger percent changes for the segmentation accuracy metrics.

Table 2.6 Highlighted percent changes in dice score from segmentations without noise removal with DBSCAN to segmentations with noise removal by DBSCAN for the complete scans.

Training View(s)	Prediction View(s)	Plant	Precision	Recall	Dice	IoU
Z	Z	Poinsettia1	-0.255	-0.918	-0.579	-1.044
Z	Z	Onion1	4.615	-0.674	1.924	3.615
Z	Z	Poinsettia2	1.444	-0.122	0.686	1.299
Z	Z	Onion2	9.023	-1.072	4.196	7.837
Average			3.707	-0.696	1.557	2.927

Table 2.7 Highlighted percent change in dice score from segmentations without noise removal with DBSCAN to segmentations with noise removal by DBSCAN for the truncated scans.

Training View(s)	Prediction View(s)	Plant	Precision	Recall	Dice	IoU
Z	Z	Poinsettia1	-1.055	-3.643	-2.331	-3.963
Z	Z	Onion1	4.832	-8.929	-0.832	-1.240
Z	Z	Poinsettia2	5.025	-0.660	2.342	4.244
Z	Z	Onion2	20.766	-2.437	10.705	18.133
Average			7.392	-3.917	2.471	4.293

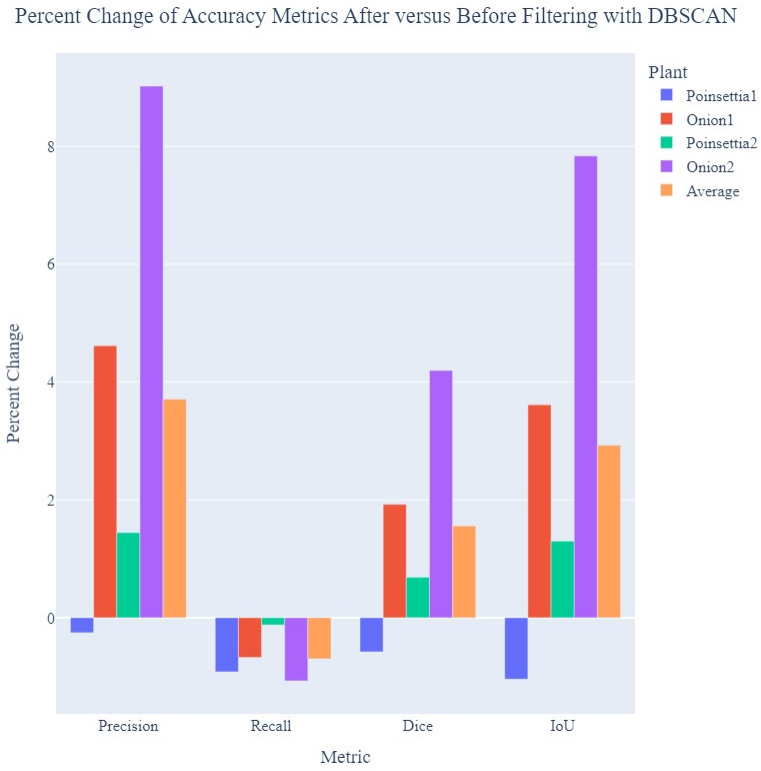


Figure 2.7 Percent change in dice score from segmentations without noise removal with DBSCAN to segmentations with noise removal by DBSCAN for predictions made on the z view with models trained on the z view for the complete scans.

Figure 2.8 Percent change in dice score from segmentations without noise removal with DBSCAN to segmentations with noise removal by DBSCAN for predictions made on the z view with models trained on the z view for the truncated scans.

Table 2.8 displays the execution times for both prediction (instance segmentation) and filtering. The maximum run time on the computer used for experimentation was less than 23 minutes. Note, these results are for the complete scans and runtime will increase with an increase in soil column dimensions. It is possible to remove images with non-root plant material or no plant material from the top and the bottom of the CT image stack before prediction to lower the execution time.

Table 2.8 Execution times for two-dimensional instance segmentation and noise removal with DBSCAN for models trained with five percent of training data.

				Instance Segmentation		Filtering with DBSCAN		Total
Training View(s)	**Prediction View(s)**	**Plant**	**Dimensions**	**Total Images**	**Time (min)**	**Points**	**Time (min)**	**Time (min)**
Z	Z	Poinsettia1	920x920x1391	1391	14.3	1.43E+07	1.5	15.8
Z	Z	Onion1	920x920x1998	1998	19.9	4.85E+06	0.9	20.7
Z	Z	Poinsettia2	920x920x1449	1449	14.4	6.22E+06	0.9	15.3
Z	Z	Onion2	920x920x1998	1998	20.4	1.87E+07	2.3	22.8

Figure 2.9, Figure 2.10, Figure 2.11, and Figure 2.12 display the ground truth, results from instance segmentation with Mask R-CNN, and results from instance segmentation with Mask R-CNN followed by noise removal with DBSCAN for Poinsetttia1, Onion1, Poinsettia2, and Onion2, respectively. It is important to note that the ground truth images were manually annotated, introducing noise into the ground truth segmentations. Therefore, “ground truth” does not mean the segmentations did not contain noise. On the other hand, it is also possible that the ground truth segmentations miss root material. This is highlighted by the segmentations of Onion2. Figure 2.12.F. contains segmented material not present in the ground truth segmentation, however, this material is consistent in size, trajectory, and continuity with the larger root system. It is likely these segmentations are from fine roots difficult to detect and track by the human annotator. Furthermore, it is important to note that the segmentations of root material differ from the complete (top) and truncated scans (bottom). The scans were truncated

prior to performing noise removal. Therefore, the value computed for ε varied for the complete and truncated scans, causing the segmentations to differ. That is to say, the truncated segmentations with filtering by DBSCAN are not truncated versions of the complete segmentations with filtering by DBSCAN.

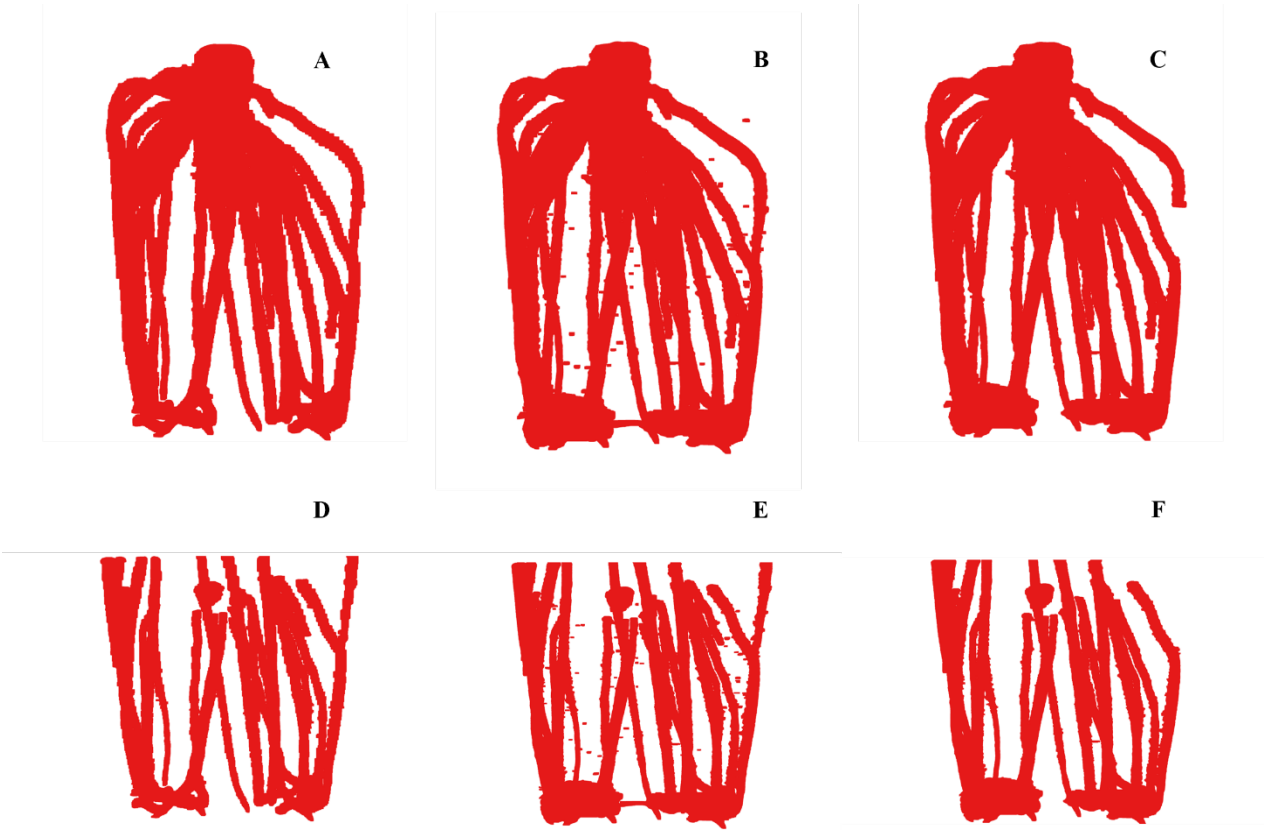


Figure 2.9 Poinsettia1 ground truth and predictions: (A) ground truth complete, (B) instance segmentation complete, (C) instance segmentation and filtering complete, (D) ground truth truncated, (E) instance segmentation truncated, (F) instance segmentation and filtering truncated.

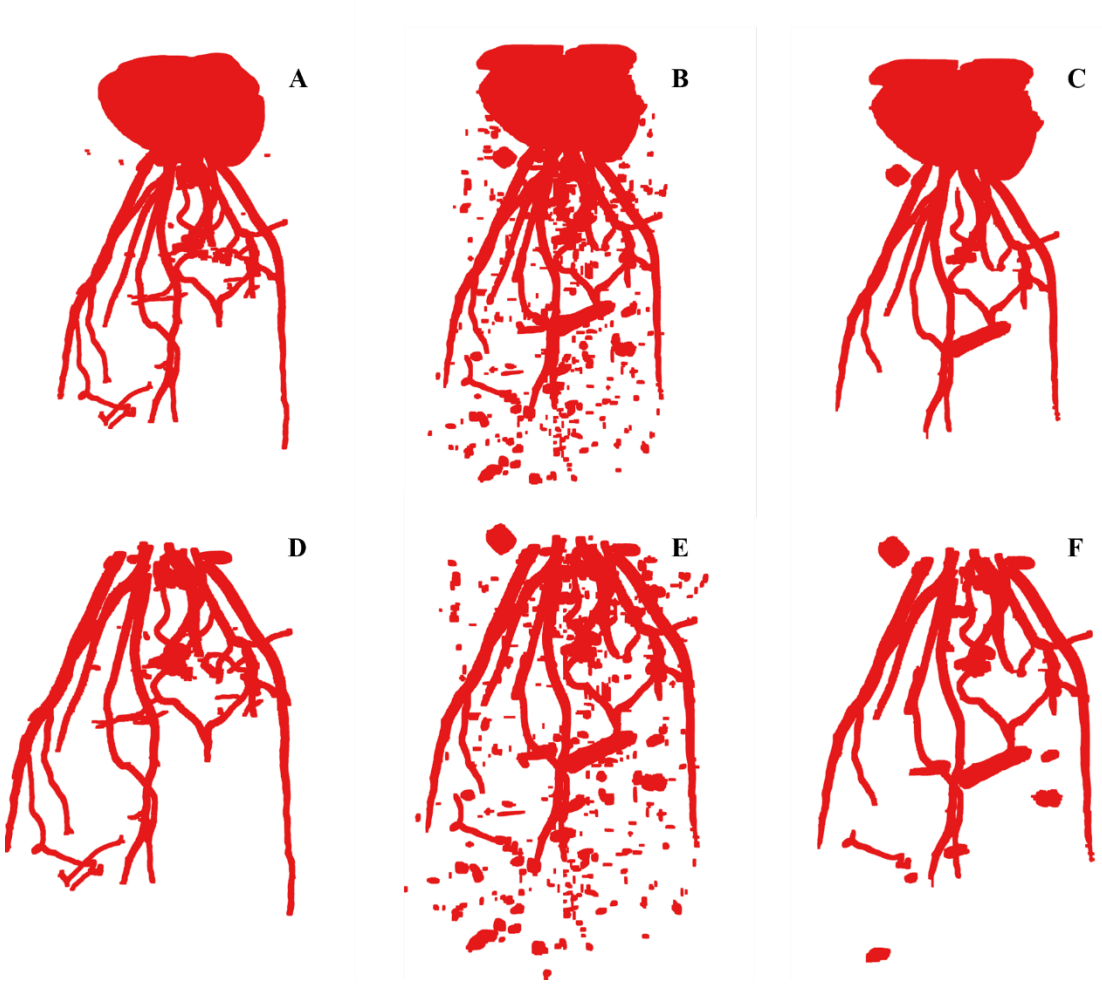


Figure 2.10 Onion1 ground truth and predictions: (A) ground truth complete, (B) instance segmentation complete, (C) instance segmentation and filtering complete, (D) ground truth truncated, (E) instance segmentation truncated, (F) instance segmentation and filtering truncated.

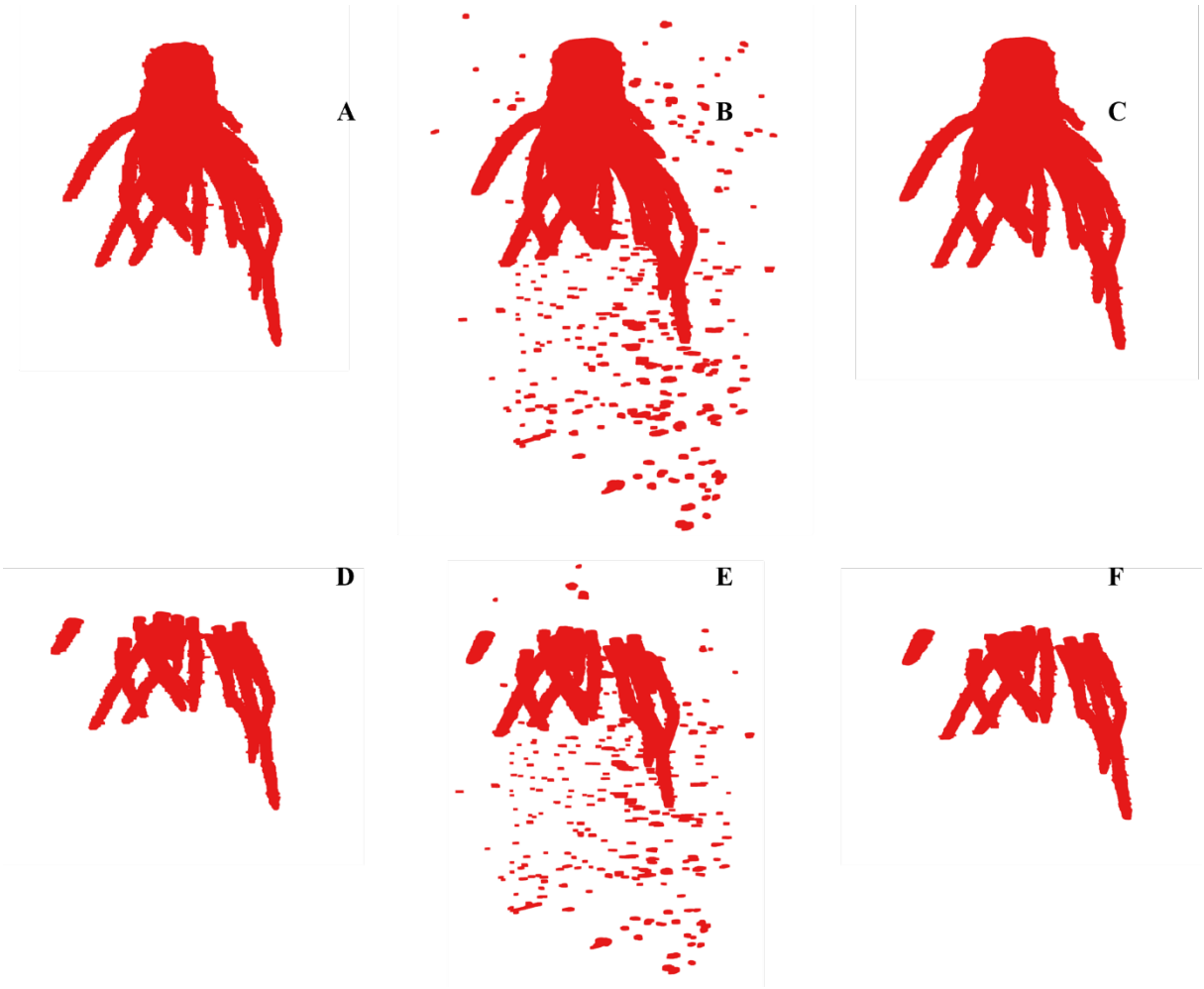


Figure 2.11 Poinsettia2 ground truth and predictions: (A) ground truth complete, (B) instance segmentation complete, (C) instance segmentation and filtering complete, (D) ground truth truncated, (E) instance segmentation truncated, (F) instance segmentation and filtering truncated.

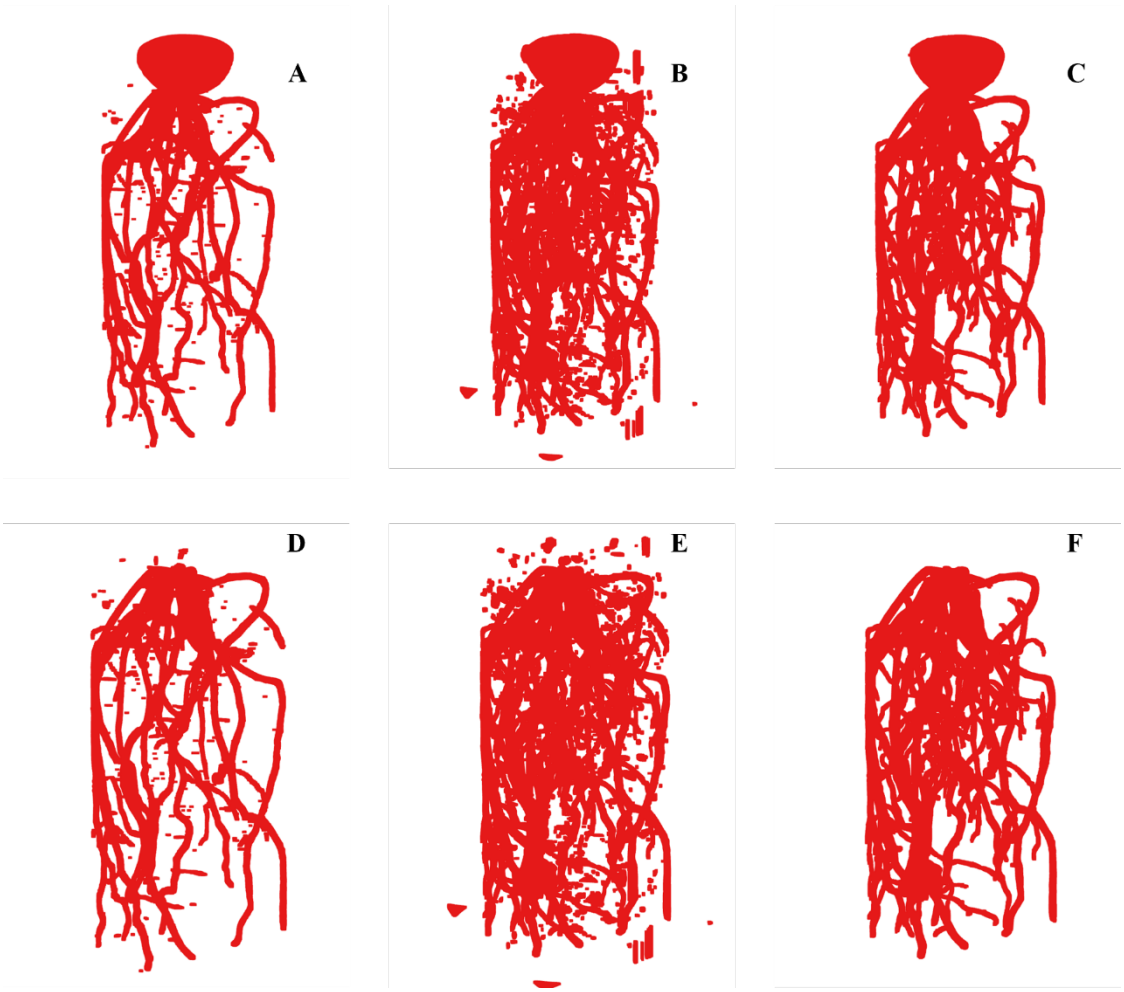


Figure 2.12 Onion2 ground truth and predictions: (A) ground truth complete, (B) instance segmentation complete, (C) instance segmentation and filtering complete, (D) ground truth truncated, (E) instance segmentation truncated, (F) instance segmentation and filtering truncated.

Table 2.9 presents results for the global thresholding. The highest average dice score occurred with a threshold of 155 to 215 for a dice score of 0.253. The ideal lower threshold ranged from 145 to 195 and the ideal upper threshold ranged from 195 to 250.

Table 2.9 Segmentation accuracy metrics obtained by global thresholding for the truncated scans.

Plant	Lower Threshold	Upper Threshold	Precision	Recall	Dice	IoU
Poinsettia1	145	195	0.412	0.701	0.519	0.350
Onion1	195	250	0.106	0.567	0.179	0.098
Poinsettia2	170	215	0.299	0.588	0.397	0.247
Onion2	185	210	0.086	0.451	0.145	0.078
Average	155	215	0.182	0.646	0.253	0.155

Figure 2.13, Figure 2.15, Figure 2.17, and Figure 2.19 display the ground truth, results from instance segmentation with Mask R-CNN followed by noise removal with DBSCAN, and results for global thresholding for Poinsetttia1, Onion1, Poinsettia2, and Onion2, respectively. The pot wall was segmented in Figure 2.13 (Poinsettia1), Figure 2.17 (Poinsettia2), and Figure 2.19 (Onion2). Figure 2.14, Figure 2.16, Figure 2.18, and Figure 2.20 display 2D sample images and binary mask predictions for instance segmentation with Mask R-CNN followed by noise removal with DBSCAN and global thresholding for Poinsetttia1, Onion1, Poinsettia2, and Onion2, respectively.

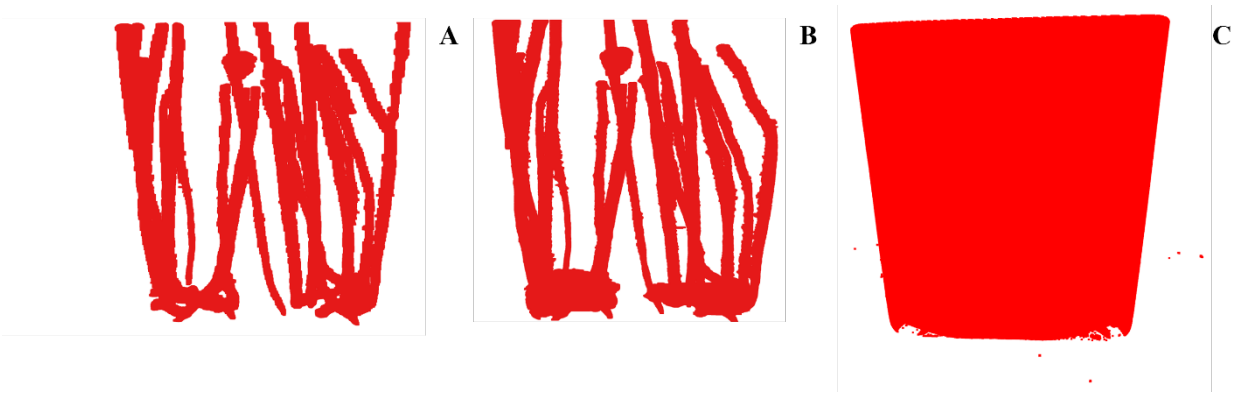


Figure 2.13 Poinsettia1 ground truth and predictions: (A) ground truth truncated, (B) instance segmentation and filtering truncated, (C) global thresholding truncated. Note, C is not to scale with A and B.

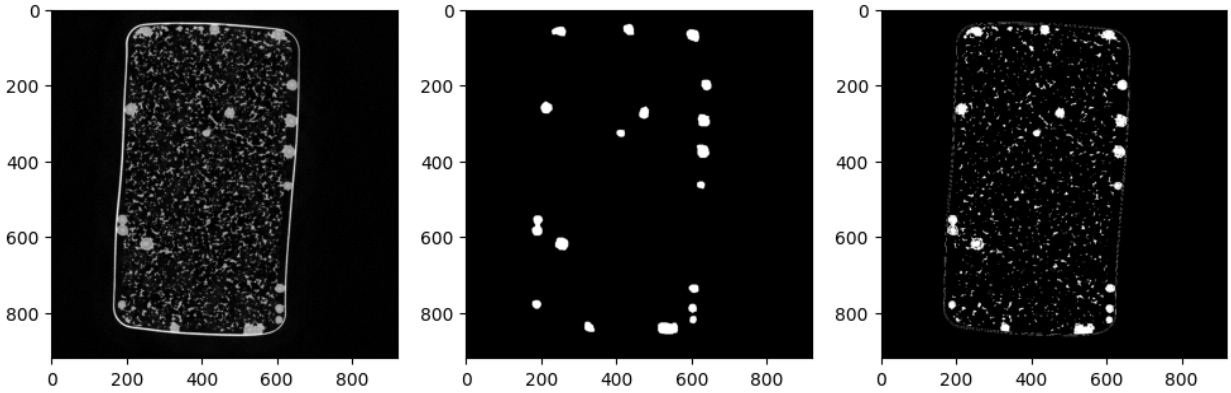


Figure 2.14 Sample images and binary mask predictions from Poinsettia1: original image (left), instance segmentation and filtering (middle), global thresholding (right).

Figure 2.15 Onion1 ground truth and predictions: (A) ground truth truncated, (B) instance segmentation and filtering truncated, (C) global thresholding truncated. Note, C is not to scale with A and B.

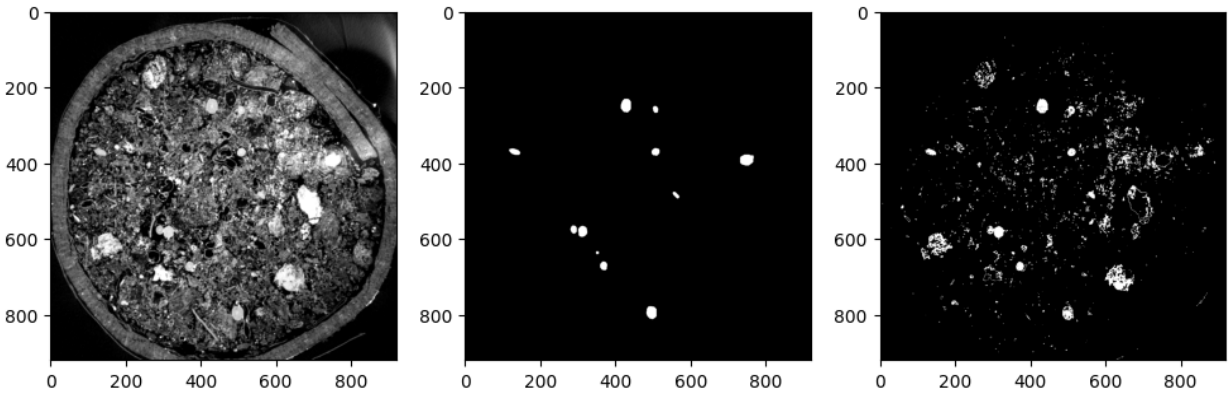


Figure 2.16 Sample images and binary mask predictions from Onion1: original image (left), instance segmentation and filtering (middle), global thresholding (right).

Figure 2.17 Poinsettia2 ground truth and predictions: (A) ground truth truncated, (B) instance segmentation and filtering truncated, (C) global thresholding truncated. Note, C is not to scale with A and B.

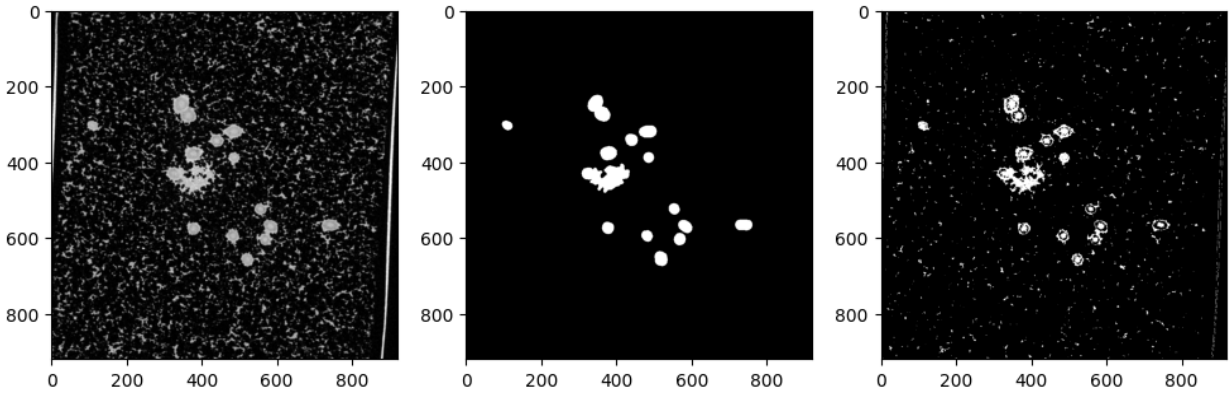


Figure 2.18 Sample images and binary mask predictions from Poinsettia2: original image (left), instance segmentation and filtering (middle), global thresholding (right).

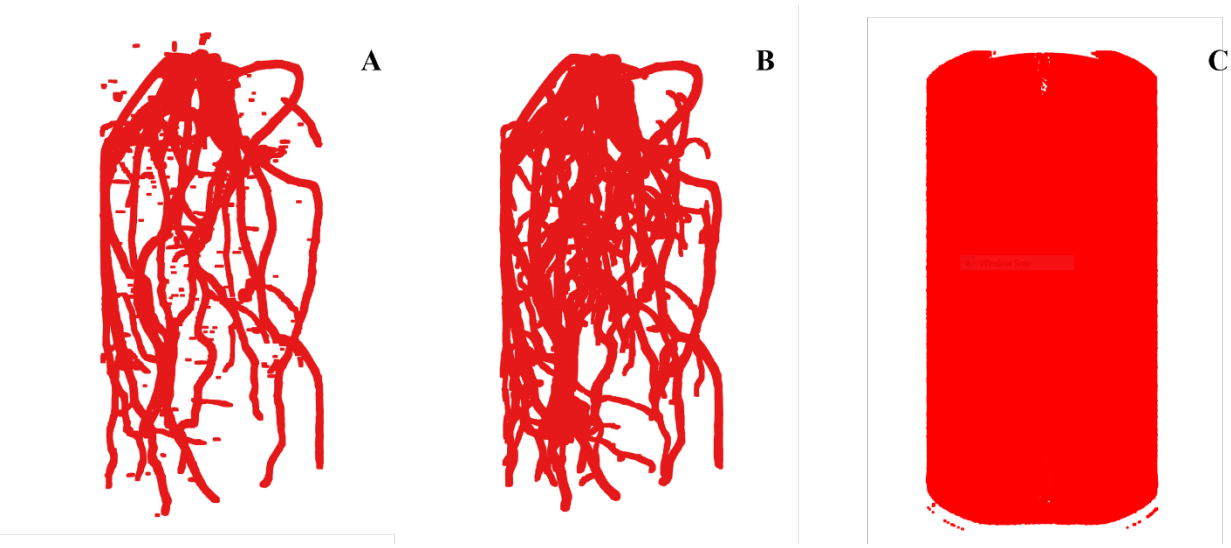


Figure 2.19 Onion2 ground truth and predictions: (A) ground truth truncated, (B) instance segmentation and filtering truncated, (C) global thresholding truncated. Note, C is not to scale with A and B.

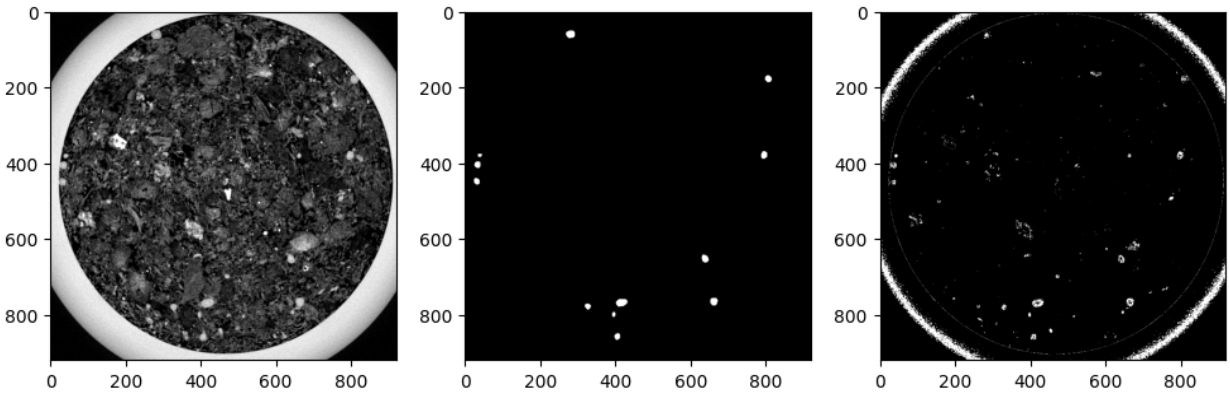
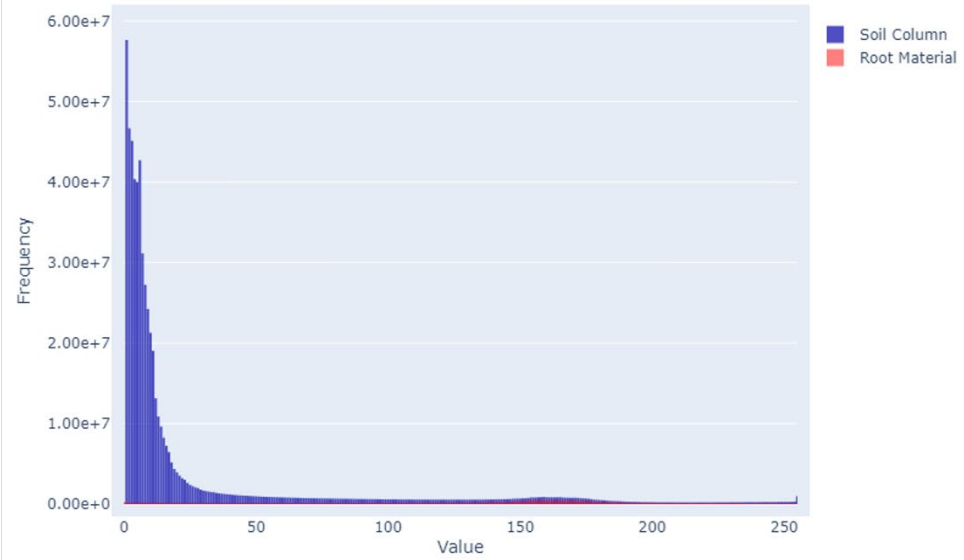


Figure 2.20 Sample images and binary mask predictions from Onion2: original image (left), instance segmentation and filtering (middle), global thresholding (right).

To corroborate the low accuracy metrics for global thresholding were incurred due to overlapping attenuation values between non-root material in the soil column and the root material, histograms of the attenuation values of the entire soil column (including the root material) and the root material were compared (Figure 2.21, Figure 2.22, Figure 2.23, and Figure

2.24). The bin with the highest frequency was removed from the entire soil column histogram for viewability, however, the histograms still demonstrate overlapping attenuation values for root and non-root material. Note, the histograms for root material (represented in red in the figures) corroborates the empirical ranges for global thresholding presented in Table 2.9

A



B

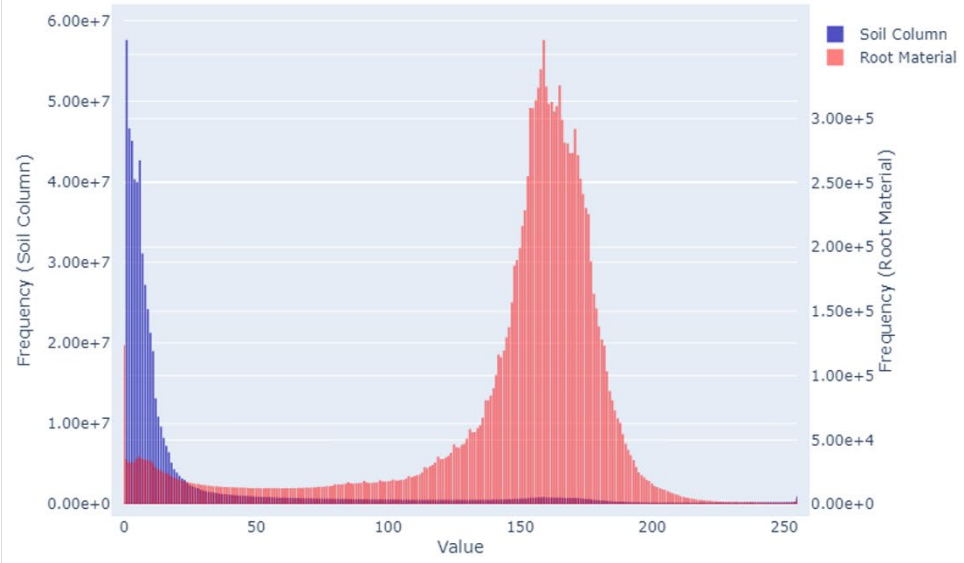
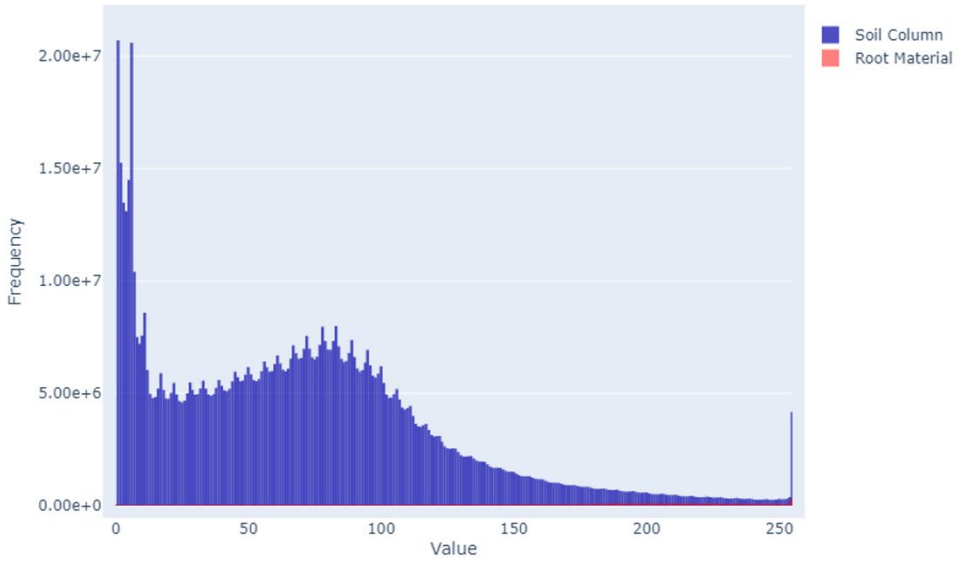


Figure 2.21 Attenutation value histograms for the the entire soil column for Poinsettia1 (including the root material) and the root material: (A) shared y-axis, (B) separate y-axis. The empirically determined ideal range for Poinsettia1 was 145 to 195 (Table 2.9).

A



B

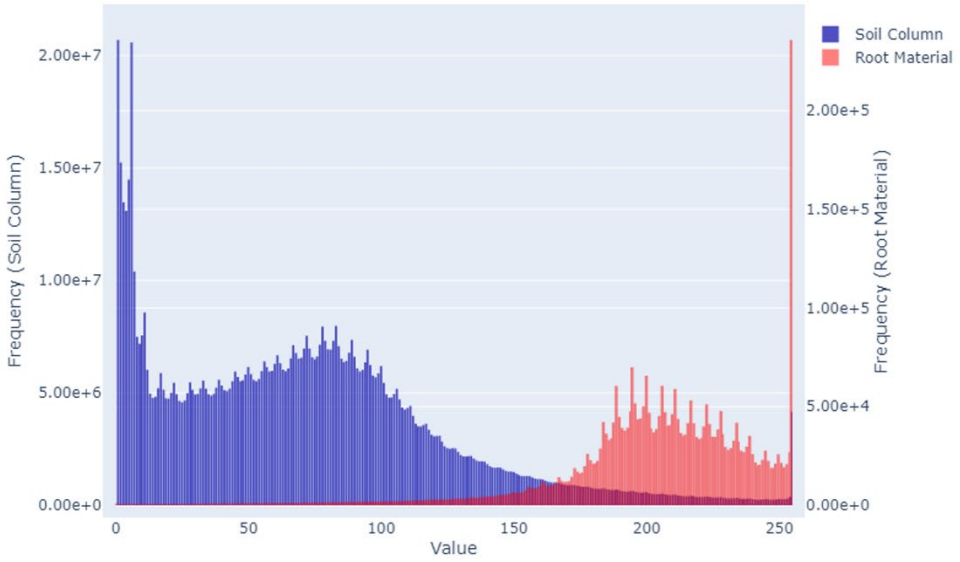
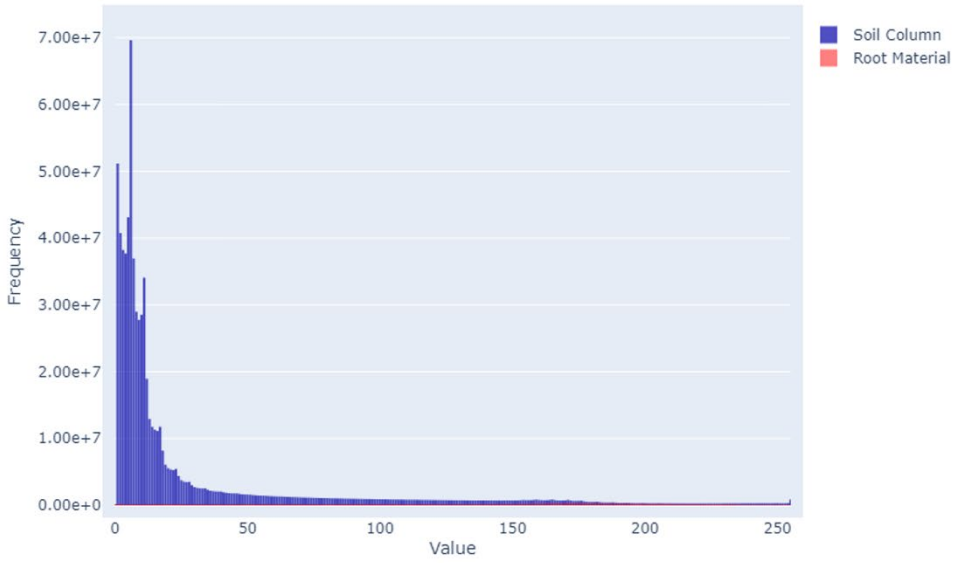


Figure 2.22 Attenutation value histograms for the the entire soil column for Onion1 (including the root material) and the root material: (A) shared y-axis, (B) separate y-axis. The empirically determined ideal range for Onion1 was 195 to 250 (Table 2.9).

A



B

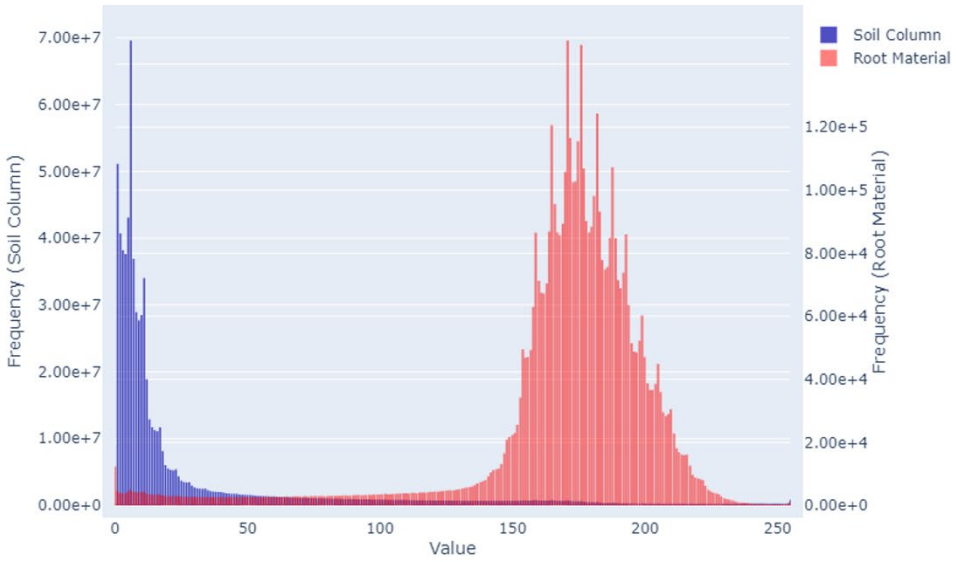
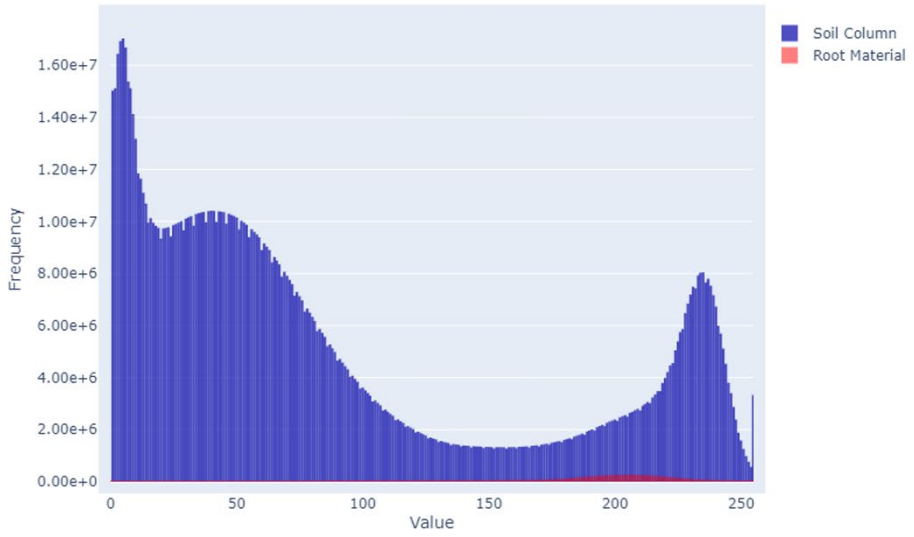


Figure 2.23 Attenutation value histograms for the the entire soil column for Poinsettia2 (including the root material) and the root material: (A) shared y-axis, (B) separate y-axis. The empirically determined ideal range for Poinsettia2 was 170 to 215 (Table 2.9).

A



B

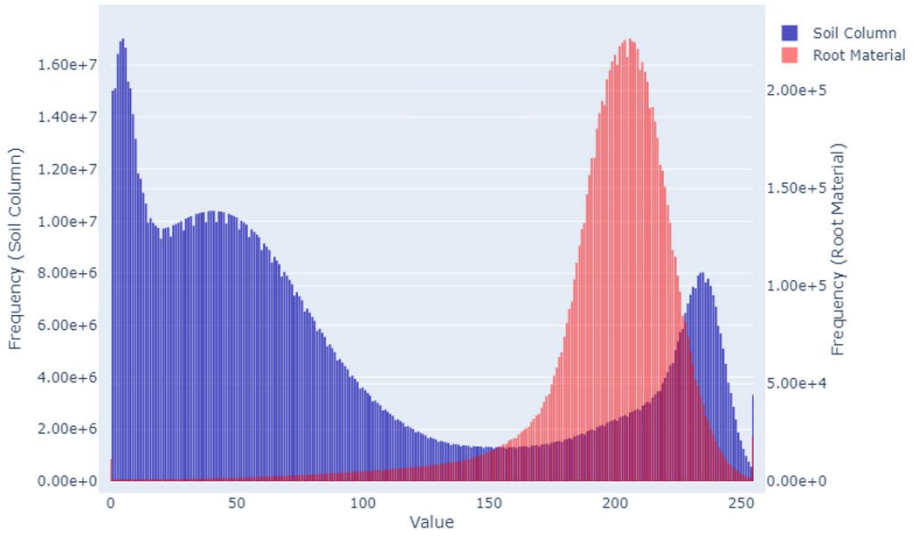


Figure 2.24 Attenutation value histograms for the the entire soil column for Onion2 (including the root material) and the root material: (A) shared y-axis, (B) separate y-axis. The empirically determined ideal range for Onion2 was 185 to 210 (Table 2.9).

Table 2.10 presents a summary of results from this work compared with the results from the previous deep learning-based approach presented in the Introduction of this paper (mutli-loss multi-resolution encoder-decoder). Note, the scores for our method (denoted by Mask R-CNN + DBSCAN) are from the truncated scans. These results indicate our method rivals the segmentation accuracies achieved by approach presented by Soltaninejad et al. However, it should be noted that claiming better segmentation accuracy for all possible plants and growth mediums combinations would require testing both methods on a common dataset with diverse samples.

Table 2.10 Summary of results and comparison to previous work.

Method	Precision	Recall	Dice	IoU
Multi-loss multi-resolution encoder-decoder	0.733	0.750	0.740	0.588
Mask R-CNN + DBSCAN, trained on xyz view with 100% of dataset	**0.831**	0.839	**0.834**	**0.718**
Mask R-CNN + DBSCAN, trained on z view with 1% of dataset	0.734	**0.868**	0.749	0.669

2.5 Discussion

The study presents an efficient and accurate method for root segmentation in CT images using deep learning-based root instance segmentation in conjunction with DBSCAN-based filtering. The proposed method achieved high accuracy scores for precision, recall, dice, and IoU, making it a promising approach for in situ RSA characterization. By utilizing 2D instance segmentation with transfer learning using a pretrained model, we capitalized on the benefits of automated feature selection by deep neural networks with the resource efficiency of image processing-based approaches. Moreover, we demonstrated that high dice scores can be achieved with as little as 1% of training data (Table 2.5).

Another notable aspect of the proposed method is the use of DBSCAN-based filtering with automated parameter tuning. Automated parameter tuning eliminated the need for human interaction that is commonly seen in image processing-based approaches. Furthermore, manual parameter tuning can be time-consuming and subjective. However, it should be noted that the noise removal implementation will always remove some of the segmentation. For segmentations that do not contain a lot of noise (such as Poinsettia1) noise removal may end up removing root material (Figure 2.9.D and 2.9.F) resulting in a decrease in dice or IoU. The user can forgo noise removal if the segmentation qualitatively appears to contain a small amount of noise. Altnernatively, an automated check for whether or not further noise removal is needed can be implemented.

We tested using different view combinations (eight total) for training and prediction with the Mask R-CNN model. We were able to conclude that training and predicting on the z view of

the soil column achieved the highest mean dice score and lowest standard deviation across different percentages of the training dataset. We also provided results for global thresholding, demonstrating thresholding alone is not an accurate method for segmenting roots from soil due to overlapping attenuation values. Automated pot wall detection could be implemented; however, this would not prevent noise introduced by the medium, evident in the 2D images (Figure 2.14, Figure 2.16, Figure 2.18, and Figure 2.20).

Overall, the proposed method presents a promising approach for accurate and efficient root segmentation from CT images. The use of deep learning-based instance segmentation and DBSCAN-based filtering can improve the accuracy of volumetric segmentation while reducing the annotation and resource requirements. The proposed method can be applied to a wide range of plant species and growth conditions as well, providing a valuable tool for studying in situ plant root system architecture. The proposed method can also be trained and used for prediction on multiple species of plants, meaning the features and parameters do not have to be selected and optimized for each plant species.

Further work could include the comparison of common phenotyping measures (e.g. length, branching, diameter, and biomass) derived from hand measurements and computed measurements using the in situ segmentation procedure presented in this work. Second, other methods such as the ones introduced in the Introduction of this chapter could be implemented on our dataset. Alternatively, our method could be used to segment roots from the datasets used for developing and testing previous methods. A compilation of a large and diverse common dataset is integral to the development of segmentation methods and the comparison of segmentation accuracies across various methods.

2.6 Conclusion

This project developed and evaluated a method for segmenting in situ roots from X-ray CT images of container horticultural plants using instance segmentation with Mask R-CNN and DBSCAN-based 3D point cloud clustering. We concluded that training and predicting on the z view achieved the highest mean dice score across different training dataset sizes. This method achieved scores of 0.734, 0.868, 0.749, and 0.669 for precision, recall, dice, and IoU respectively utilizing only 1% of the training dataset (34 CT images for Dataset1 and 35 CT images for Datset2). This method allows for the use of deep learning and machine learning to automate RSA segmentation without the burden of large GPU requirements or manually annotating large datasets for training.

2.7 Acknowledgments

This project was supported by the Alabama Agricultural Experiment Station. The authors would like to thank Spencer Overton and Austin Lindquist for supporting this project by performing manual annotation of the datasets.

2.8 References

Bradski, G. (2000). *The OpenCV Library*. Dr. Dobb's Journal of Software Tools.

Brooks, J. (2019). *COCO Annotator*. https://github.com/jsbroks/coco-annotator

Downie, H. F., Adu, M. O., Schmidt, S., Otten, W., Dupuy, L. X., White, P. J., & Valentine, T. A. (2015). Challenges and opportunities for quantifying roots and rhizosphere interactions

through imaging and image analysis. *Plant, Cell & Environment*, *38*(7), 1213–1232. https://doi.org/10.1111/pce.12448

Flavel, R. J., Guppy, C. N., Rabbi, S. M. R., & Young, I. M. (2017). An image processing and analysis tool for identifying and analysing complex plant root systems in 3D soil using non-destructive analysis: Root1. *PloS one*, *12*(5), e0176433. https://doi.org/10.1371/journal.pone.0176433

Frangi, A. F., Niessen, W. J., Vincken, K. L., & Viergever, M. A. (1998). Multiscale vessel enhancement filtering. In *Medical Image Computing and Computer-Assisted Intervention—MICCAI'98: First International Conference Cambridge, MA, USA, October 11–13, 1998 Proceedings 1* (pp. 130-137). Springer Berlin Heidelberg. https://doi.org/10.1007/BFb0056195

Gaggion, N., Ariel, F., Daric, V., Lambert, É., Legendre, S., Roulé, T., Camoirano, A., Milone, D. H., Crespi, M., Blein, T., & Ferrante, E. (2021). ChronoRoot: High-throughput phenotyping by deep segmentation networks reveals novel temporal parameters of plant root system architecture. *GigaScience*. https://doi.org/10.1093/gigascience/giab052

Galkovskyi, T., Mileyko, Y., Bucksch, A., Moore, B., Symonova, O., Price, C. A., Topp, C. N., Iyer-Pascuzzi, A. S., Zurek, P. R., Fang, S., Harer, J., Benfey, P. N., & Weitz, J. S. (2012). GiA Roots: software for the high throughput analysis of plant root system architecture. *BMC Plant Biology*, *12*(1), 116. https://doi.org/10.1186/1471-2229-12-116

Gao, W., Schlüter, S., Blaser, S. R. G. A., Shen, J., & Vetterlein, D. (2019). A shape-based method for automatic and rapid segmentation of roots in soil from X-ray computed

tomography images: Rootine. *Plant and Soil*, *441*(1–2), 643–655. https://doi.org/10.1007/s11104-019-04053-6

Gerth, S., Claußen, J., Eggert, A., Wörlein, N., Waininger, M., Wittenberg, T., & Uhlmann, N. (2021). Semiautomated 3D root segmentation and evaluation based on X-ray CT imagery. *Plant Phenomics*, *2021*. https://doi.org/10.34133/2021/8747930

He, K., Gkioxari, G., Dollar, P., & Girshick, R. (2020). Mask R-CNN. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *42*(2), 386–397. https://doi.org/10.1109/TPAMI.2018.2844175

Janzen, H. (2002). Correcting mathematically for soil adhering to root samples. *Soil Biology and Biochemistry*, *34*(12), 1965–1968. https://doi.org/10.1016/S0038-0717(02)00206-7

Kang, J., Liu, L., Zhang, F., Shen, C., Wang, N., & Shao, L. (2021). Semantic segmentation model of cotton roots in-situ image based on attention mechanism. *Computers and Electronics in Agriculture*, *189*, 106370. https://doi.org/10.1016/j.compag.2021.106370

Koevoets, I. T., Venema, J. H., Elzenga, J. Theo. M., & Testerink, C. (2016). Roots withstanding their environment: exploiting root system architecture responses to abiotic stress to improve crop tolerance. *Frontiers in Plant Science*, *07*. https://doi.org/10.3389/fpls.2016.01335

Lin, J. (1991). Divergence measures based on the Shannon entropy. *IEEE Transactions on Information Theory*, *37*(1), 145–151. https://doi.org/10.1109/18.61115

Livesley, S. J. (1998). Sieve size effects on root length and biomass measurements of maize (Zea mays) and Grevillea robusta. *Plant and Soil*, *207*(2), 183–193. https://doi.org/10.1023/A:1026461107110

Mairhofer, S., Zappala, S., Tracy, S. R., Sturrock, C., Bennett, M., Mooney, S. J., & Pridmore, T. (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. https://doi.org/10.1104/pp.111.186221

Mooney, S. J., Pridmore, T. P., Helliwell, J., & Bennett, M. J. (2012). Developing X-ray Computed Tomography to non-invasively image 3-D root systems architecture in soil. *Plant and Soil*, *352*(1–2), 1–22. https://doi.org/10.1007/s11104-011-1039-9

Phalempin, M., Lippold, E., Vetterlein, D., & Schlüter, S. (2021). An improved method for the segmentation of roots from X-ray computed tomography 3D images: Rootine v.2. *Plant Methods*, *17*, 39. https://doi.org/10.1186/s13007-021-00735-4

Sander, J., Ester, M., Kriegel, H.-P., & Xu, X. (1998). Density-based clustering in spatial databases: The algorithm gdbscan and its applications. *Data Mining and Knowledge Discovery*, *2*(2), 169–194. https://doi.org/10.1023/A:1009745219419

Satopaa, V., Albrecht, J., Irwin, D., & Raghavan, B. (2011). Finding a" kneedle" in a haystack: Detecting knee points in system behavior. *2011 31st International Conference on Distributed Computing Systems Workshops*, 166–171. https://doi.org/10.1109/ICDCSW.2011.20

Schubert, E., Sander, J., Ester, M., Kriegel, H. P., & Xu, X. (2017). DBSCAN revisited, revisited: why and how you should (still) use DBSCAN. *ACM Transactions on Database Systems*, *42*(3), 1–21. https://doi.org/10.1145/3068335

Sethian, J. A. (1999). *Level Set Methods and Fast Marching Methods: Evolving Interfaces in Computational Geometry, Fluid Mechanics, Computer Vision, and Materials Science* (2nd ed.). Cambridge University Press.

Shao, M. R., Jiang, N., Li, M., Howard, A., Lehner, K., Mullen, J. L., Gunn, S. L., McKay, J. K., & Topp, C. N. (2021b). Complementary phenotyping of maize root system architecture by root pulling force and X-ray imaging. *Plant Phenomics*, *2021*. https://doi.org/10.34133/2021/9859254

Shen, C., Liu, L., Zhu, L., Kang, J., Wang, N., & Shao, L. (2020). High-throughput in situ root image segmentation based on the improved DeepLabv3+ method. *Frontiers in Plant Science*, *11*. https://doi.org/10.3389/fpls.2020.576791

Smith, S., & De Smet, I. (2012). Root system architecture: insights from Arabidopsis and cereal crops. *Philosophical Transactions of the Royal Society B: Biological Sciences*, *367*(1595), 1441–1452. https://doi.org/10.1098/rstb.2011.0234

Soltaninejad, M., Sturrock, C. J., Griffiths, M., Pridmore, T. P., & Pound, M. P. (2020). Three dimensional root CT segmentation using multi-resolution encoder-decoder networks. *IEEE Transactions on Image Processing*, *29*, 6667–6679. https://doi.org/10.1109/TIP.2020.2992893

Tabb, A., Duncan, K. E., & Topp, C. N. (2018). Segmenting root systems in X-ray computed tomography images using level sets. *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*, 586–595. https://doi.org/10.1109/WACV.2018.00070

Thesma, V., & Mohammadpour Velni, J. (2022). Plant root phenotyping using deep conditional GANs and binary semantic segmentation. *Sensors*, *23*(1), 309. https://doi.org/10.3390/s23010309

Tollner, E. W. (1991). X-ray computed tomography applications in soil ecology studies. *Agriculture, Ecosystems & Environment*, *34*(1–4), 251–260. https://doi.org/10.1016/0167-8809(91)90112-B

van Noordwijk, M., & Floris, J. (1979). Loss of dry weight during washing and storage of root samples. *Plant and Soil*, *53*(1–2), 239–243. https://doi.org/10.1007/BF02181896

van Rossum, G. (2022). *Python 3 Reference Manual*. https://docs.python.org/release/3.10.9/

Wang, T., Rostamza, M., Song, Z., Wang, L., McNickle, G., Iyer-Pascuzzi, A. S., Qiu, Z., & Jin, J. (2019). SegRoot: A high throughput segmentation method for root image analysis. *Computers and Electronics in Agriculture*, *162*, 845–854. https://doi.org/10.1016/j.compag.2019.05.017

Wu, Y., Kirillov, A., Massa, F., Lo, W.-Y., & Girshick, R. (2019). *Detectron2*. https://github.com/facebookresearch/detectron2

Yu, Q., Tang, H., Zhu, L., Zhang, W., Liu, L., & Wang, N. (2023). A method of cotton root segmentation based on edge devices. *Frontiers in Plant Science*, *14*. https://doi.org/10.3389/fpls.2023.1122833

Zhou, Q.-Y., Park, J., & Koltun, V. (2018). *Open3D: A Modern Library for 3D Data Processing*. http://www.open3d.org/

Chapter 3. High-throughput phenotyping of *Humulus lupulus* in greenhouse using stereo vision, semantic segmentation, and 3D point cloud analysis

3.1 Abstract

Humulus lupulus (hops) is an important crop, both politically and economically (Legun et al., 2022). The USA is a major producer of hops, sharing 75-80% of world hop production with Germany (Almaguer et al., 2014)**.** One of the major factors affecting hop quality is environmental conditions (Neve, 1991). Therefore, there is interest in breeding hop cultivars for specific geographic environments. Conducting variety trials offers a tool for testing hop cultivars suitable for various environments. However, conducting variety trials relies on collecting large amounts of phenotypic data, including morphological data. Computer vision and machine learning technologies equip researchers with advanced capabilities for conducting high-throughput experiments for the measurement of phenotypic traits in variety trials. However, limited research has been conducted for the application of computer vision and machine learning to high-throughput phenotyping for vine crops, especially morphological analysis, and even less research has been conducted developing pipelines for hop phenotyping. In this study, an efficient method for measuring vine length, leaf area, and biomass from three-dimensional (3D) point clouds was developed and evaluated for hops grown in greenhouse. Point clouds were retrieved from ZED 2 stereo cameras for 3D scene reconstruction. Additionally, a two-dimensional (2D) SegFormer model was fine-tuned on images for semantic segmentation of hop

plants from a greenhouse scene for morphological trait measurement. Measurements of vine length, leaf area, and biomass were derived from the segmented point clouds, yielding high R^2 values of 0.79, 0.95, and 0.91, respectively, indicative of strong correlation between the derived measurements and the ground truth measurements.

3.2 Introduction

The written record of use of hops for brewing dates to 1963, found in the Finnish work *The Kalevala* by Lönnrot, though the written record for hop cultivation predates this by 1227 years, as recorded in Germany in 736 AD (Neve, 1991). Today, about 97% of hop production is used for brewing (Almaguer et al., 2014; Schönberger & Kostelecky, 2011) with Germany and the USA producing 75-80% of hops (Almaguer et al., 2014). According to the National Agricultural Statistics Service, the production value of hops in the USA was over 617 million dollars in 2022. Legun et al. deem hops a *neoliberal crop*, shaping both politics and economies globally, credited to the emerging practice of breeding hops for both flavor and aroma in addition to bitterness (2022). Hops that are cultivated for their flavor as opposed to bitterness are called *aroma hops* and are essential to craft beer production, an industry dominated by the USA (Knudson et al., 2020; Legun et al., 2022). Although hops are largely cultivated today for their bitterness and flavoring, their first use was for their medicinal and therapeutic properties, most often as mild sedatives (Turner et al., 2011). The use of hops for medicinal applications is reflected in contemporary research today (J.S. Bland et al., 2015; Karabín et al., 2016; Wang et al., 2012; Zanoli et al., 2005).

Interest in hop cultivation is attributed to the unique composition of their lupulin gland and the resins it produces, which are not found in any other plant species (Neve, 1991). Numerous interacting factors influence the quality and yield of the resin, affecting the desirability of the hop yield. These factors range from mitigating pest and disease occurrences to adjusting nutrient applications, with variations in environmental conditions such as temperature, rainfall, and sunlight also playing substantial roles. However, the magnitude of the effect of each factor and the interplay between these factors is contested, although it is likely the largest factor in hop health and yield is environmental factors, including geographic location (Neve, 1991).

Because of their need to climb as they grow, it is common to grow hops on V-shaped trellises, with the height of the trellis limiting the height of hop vine growth. Note, the vines of hops are properly named bines, however, in this work they will be referred to colloquially as vines. The height of hop growth also affects the value of the harvest, as different varieties of hops produce their greatest yield of cones at different maximum trellis heights (Neve, 1991). Hops are typically grown between five and eight meters tall, though countries such as the USA and China typically grow hops on supports half of this height (Neve, 1991). The height of hop trellises makes hand measurements for hop phenotyping both ergonomically difficult and potentially dangerous to obtain, if standing on a raised platform or ladder is necessary to reach the uppermost end of the plant. Furthermore, the measurement of morphological traits such as leaf area and biomass often require destructive techniques, i.e., cutting the vine so that the plant can be placed in the appropriate measurement tool, meaning these measurements cannot be measured for the same plant over an extended period of time. Therefore, this work will focus on the development of an automated pipeline for a high-throughput method of hop morphological

analysis, providing steps towards both relieving the burden of the intensive process of taking hand measurements and allowing for the non-destructive measurement of leaf area and biomass over the lifespan of the plant. High-throughput tools for measuring morphological traits would assist in variety trials for determining which varieties grow best with varying factors, including geographic location, environmental conditions, and trellis length and angle.

When compared to advances in technologies for high-throughput plant genotyping, advancement in technologies for high-throughput plant phenotyping was delayed (Ninomiya, 2022; White et al., 2012). In response to this, there has been a recent surge in published studies in plant phenomics, accompanied by the start of many research centers dedicated to plant phenotyping, including the Australian Plant Phenomics Facility (AU), the Jülich Plant Phenotyping Center (DE), the National Plant Phenomics Center (UK), the Plant Phenotyping and Imaging Research Center (CA), and the Plant Phenomics Research Center (CN) (Ninomiya, 2022) and the Danforth Center (USA). The growing development of computer vision and machine learning tools, including open-source software, provides the necessary tools for high-throughput plant phenotyping, helping bridge the disparity in technologies for rapid genotyping and phenotyping. Distinct applications of computer vision and machine learning to above-ground plant phenotyping include: plant disease detection (Roy & Bhaduri, 2021; Singh et al., 2020), stress detection (Ghosal et al., 2018; Ramos-Giraldo et al., 2020), yield prediction (Liu et al., 2017; Pothen & Nuske, 2016), and morphology, including plant architecture and measurements such as leaf area and biomass (Azzari et al., 2013; Chaivivatrakul et al., 2014; Jiang et al., 2016; Niknejad et al., 2023; Xiang et al., 2019). Furthermore, recent open-source libraries have been developed for specifically for the task of phenotyping plant morphology, including: PlantCV,

developed for two-dimensional (2D) plant phenotyping, (Gehan et al., 2017) and Phenomenal, developed for both two-dimensional and three-dimensional (3D) plant phenotyping (Artzet et al., 2019).

Computer vision and machine learning has been used for assessing vine crop health using images from unmanned aerial vehicles (UAV) (Kerkech et al., 2020a, 2020b) and 3D image processing techniques such as stereo vision have been used for 3D reconstruction and object detection in vineyard applications (Nellithimaru & Kantor, 2019) and computation of fruit-to-leaf ratios in 3D reconstructions (Klodt et al., 2015). However, studies on vine morphology are not common. Even less common is the application of computer vision and machine learning to high-throughput phenotyping of *Humulus lupulus* (hops), a twinning perennial vine. Though computer vision has been applied to disease detection and variety classification for hops (Castro et al., 2022; Farhanah & Al Maki, 2022), no work has been completed for either 2D of 3D morphological analysis of hops plants. The limited progress in this area may stem from the unstructured architecture of hop plants.

The objective of this study is to develop a semi-automated approach for measuring phenotypic traits of hops, including, vine length, leaf area, and biomass. We propose fine-tuning a 2D SegFormer (Xie et al., 2021) model from the Hugging Face Transformers library (Wolf et al., 2020) for semantic segmentation of hop plant from 2D images. Stereolabs' ZED 2 (Stereolabs, San Francisco, CA) camera was used to capture ZED SVO files. Left right stereo image pairs and corresponding point clouds were retrieved from the SVO files. Hop plants were segmented from the left stereo image using the fine-tuned SegFormer model and the 2D segmentations were applied to the 3D point clouds. A semi-automated pipeline was developed

and assessed to measure vine length, leaf area, and biomass from the point clouds. The V-shaped architecture imposed on the hops by growing them on a trellis was leveraged for individual vine separation and length measurement using random sample consensus (RANSAC) (Fischler & Bolles, 1981). Furthermore, total leaf area for each plant was estimated by summing the total triangle areas in a triangle mesh of the segmented 3D point cloud. Biomass results regressed from voxel count and leaf area were compared.

To the best of our knowledge, this is the first work to employ computer vision and machine learning for automating individual plant segmentation and 3D reconstruction for morphological phenotypic trait extraction for hops.

3.3 Materials and Methods

3.3.1 Data Collection

All hop plants used in this study were grown in Patterson Greenhouse at Auburn University (Auburn, AL). There were sixteen plants used in this study and data was collected for both the North and South view of the plants seven times over a period of four weeks. Hand measurements were taken for vine length at each data collection date. On the eighth and final data collection date, biomass and total leaf area were collected for each plant. Left and right stereo image pairs were collected and stored as SVO files using a ZED 2 stereo camera mounted on a tripod. The intrinsic parameters of stereo camera used are as follows: focal length_x = 1054.760 pixels, focal length_y = 1054.470 pixels, principal point coordinates C_x = 1100.000 pixels and C_y = 651.339 pixels, and baseline T_x = 0.120 m. The images were collected at

differing times of day on each date, resulting in a wide range of lighting conditions captured. The greenhouse layout can be seen in Figure 3.1.

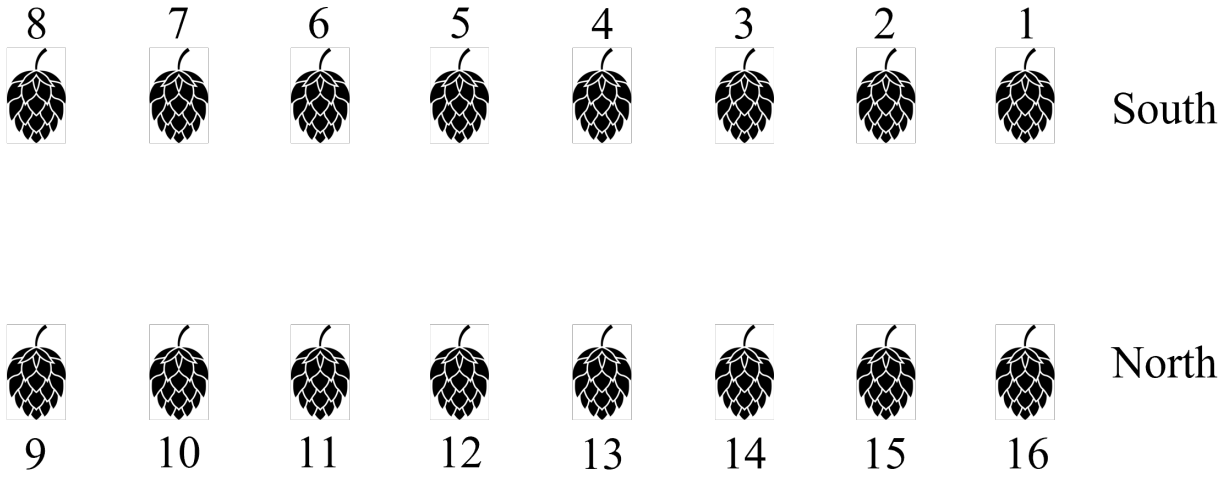


Figure 3.1 Patterson Greenhouse layout. Hop art obtained from Shutterstock.

3.3.2 Data Pre-Processing

Left and right stereo pairs were retrieved using ZED SDK (Stereolabs) and split into two datasets: Dataset1, used for training and evaluating the semantic segmentation model, and Dataset2, used for generating point clouds and measuring phenotypic traits. Dataset1 contained the South facing images of plants 1 through 8 and the North facing images of plants 9 through 16. Dataset2 contained the North facing images of plants 1 through 8 and the South facing images of plants 9 through 16. The North facing images of plant 13 retrieved on November 29, 2021 were removed from Dataset1 and included in Dataset2 to account for missing South view data on this day. Each image was 2208 × 1242 pixels large. The images from Dataset1 were tiled

into 512 × 512 pixel tiles for model training. Ground truth annotations were obtained by manual annotation of hop plants in Dataset1 using Roboflow (Dwyer et al., 2022).

Table 3.1 Data collection dates and dataset split.

Dataset	Date	Plants
1	November 12, 2021	1S, 2S, 3S, 4S, 5S, 6S, 7S, 8S, 9N, 10N, 11N, 12N, 13N, 14N, 15N, 16N
	November 17, 2021	1S, 2S, 3S, 4S, 5S, 6S, 7S, 8S, 9N, 10N, 11N, 12N, 13N, 14N, 15N, 16N
	November 19, 2021	1S, 2S, 3S, 4S, 5S, 6S, 7S, 8S, 9N, 10N, 11N, 12N, 13N, 14N, 15N, 16N
	November 23, 2021	1S, 2S, 3S, 4S, 5S, 6S, 7S, 8S, 9N, 10N, 11N, 12N, 13N, 14N, 15N, 16N
	November 29, 2021	1S, 2S, 3S, 4S, 5S, 6S, 7S, 8S, 9N, 10N, 11N, 12N, 14N, 15N, 16N
	December 03, 2021	1S, 2S, 3S, 4S, 5S, 6S, 7S, 8S, 9N, 10N, 11N, 12N, 13N, 14N, 15N, 16N
	December 09, 2021	1S, 2S, 3S, 4S, 5S, 6S, 7S, 8S, 9N, 10N, 11N, 12N, 13N, 14N, 15N, 16N
2	November 12, 2021	1N, 2N, 3N, 4N, 5N, 6N, 7N, 8N, 9S, 10S, 11S, 12S, 13S, 14S, 15S, 16S
	November 17, 2021	1N, 2N, 3N, 4N, 5N, 6N, 7N, 8N, 9S, 10S, 11S, 12S, 13S, 14S, 15S, 16S
	November 19, 2021	1N, 2N, 3N, 4N, 5N, 6N, 7N, 8N, 9S, 10S, 11S, 12S, 13S, 14S, 15S, 16S
	November 23, 2021	1N, 2N, 3N, 4N, 5N, 6N, 7N, 8N, 9S, 10S, 11S, 12S, 13S, 14S, 15S, 16S
	November 29, 2021	1N, 2N, 3N, 4N, 5N, 6N, 7N, 8N, 9S, 10S, 11S, 12S, 13N, 14S, 15S, 16S
	December 03, 2021	1N, 2N, 3N, 4N, 5N, 6N, 7N, 8N, 9S, 10S, 11S, 12S, 13S, 14S, 15S, 16S
	December 09, 2021	1N, 2N, 3N, 4N, 5N, 6N, 7N, 8N, 9S, 10S, 11S, 12S, 13S, 14S, 15S, 16S
	December 16, 2021	1N, 2N, 3N, 4N, 5N, 6N, 7N, 8N, 9S, 10S, 11S, 12S, 13S, 14S, 15S, 16S

3.3.3 2D Semantic Segmentation with SegFormer

The pretrained model and configuration was provided by the Hugging Face Transformers Library (Wolf et al., 2020). Dataset1 was split into 2235 images for training, 528 images for validation, and 480 images for testing. Random horizontal flips, random vertical flips, and random rotations between 0° and 180° were performed during the loading of the training and validation sets. The model was trained using Google Colab Pro+ with a learning rate of 0.0006 and the optimizer AdamW for 20 epochs.

The model trained on Dataset1 was used to predict the semantic segmentations of hop plants in the left images of Dataset2. During prediction, each image was tiled into 512 × 512 pixel images. The predicted masks were recombined to produce masks for the 2208 × 1242 pixel left image.

3.3.4 Computer Vison-Assisted Phenotypic Trait Measurement

To obtain measurements for length, leaf area, and biomass, 3D point clouds were retrieved from the SVO files using the Depth Sensing API from ZED SDK using the neural depth mode. Left right stereo image pairs corresponding to the point clouds were also retrieved from the SVO files. The plant was segmented from the left image and the 2D segmentation was applied to the 3D point cloud to segment the hop plant from the 3D scene. The plant of interest was isolated from the scene by a user using CloudCompare (CloudCompare, 2022). Next, the point cloud was downsampled using *voxel_down_sample* from Open3D 0.16.0 (Zhou et al., 2018) with a voxel size of 1 mm. Outliers were removed from the segmented point cloud using *statistical_outlier_removal* from Open3D with a standard deviation of 2.0 and 200 neighbors.

Vine length was obtained by fitting a line to each vine using the random sample consensus algorithm (RANSAC). The line was cropped to line points within 20 mm of RANSAC inliers and length was measured as the distance between the first and last point on the cropped line. For plants with two vines, the first vine was removed after it was measured by removing the inliers of the line within a threshold of 60 mm. The second vine could then be detected and measured with a second iteration of RANSAC. RANSAC was implemented using pyRANSAC-3D (Mariga, 2022). Plant material below the height of the pot was removed prior to fitting a line with RANSAC to avoid extra plant material not in the vine of interest skewing the fitted line. To do this, a plane was fitted to the ground in the unsegmented point cloud using Open3D's implementation of RANSAC for plane fitting, *segment_plane*. The height of the plane was adjusted to the pot height and point cloud data below the pot height at 340 mm was trimmed from both the unsegmented and segmented point cloud. After the plants reached the top of the trellis length was not measured because the vines often became inseparable at this point. Furthermore, there had to be enough upward vine growth on the trellis to reliably fit a RANSAC line for vine length measurement, especially for plants with excess vine material growing at the bottom of the plant. Figure 3.2.A (plant 10S imaged on November 29, 2021) presents a case where there was not enough upward vine growth on the trellis to reliably measure height. Figure 3.2.B (plant 10S imaged on December 09, 2021) presents a case where there was enough upward vine growth on the trellis to reliably measure height. Alternatively, the height under which point cloud data would be removed can be adjusted. However, adjusting the height can affect the length measurements for other plants.

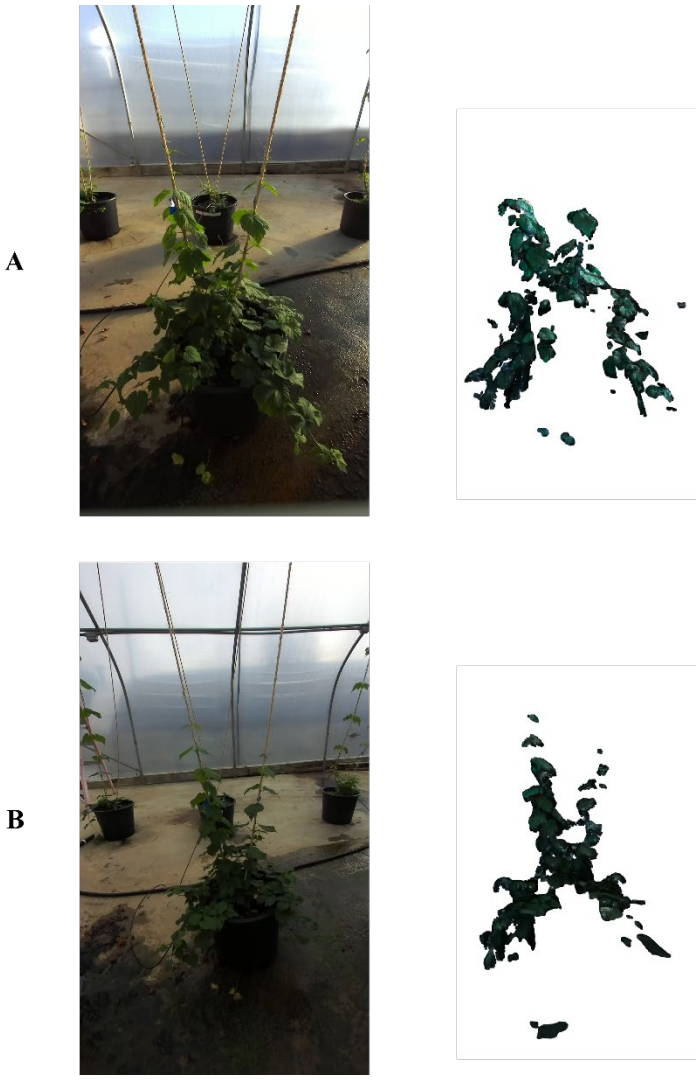


Figure 3.2 Plant 10S imaged on (A) November 29, 2021, (B) December 09, 2021 with corresponding point clouds (right).

To obtain measurements for leaf area, the ball pivoting algorithm (BPA) (Bernardini et al., 1999) implemented by Open3D was used to create a triangle mesh. The radii for the ball pivoting algorithm were determined by two methods: examining the histograms of the average distances between each point and its six nearest neighbors (Figure 3.3) and empirically by testing multiple sets of radii, including: [1] mm, [2, 1] mm, [4, 2, 1] mm, and [8, 4, 2, 1] mm. An ideal

first guess for radius for BPA occurs at the peak of the histogram. Radii below the radius indicated by this peak will be insufficient for created triangle mesh from the point cloud. Empirically, we found a radii combination of [4, 2, 1] mm to achieve the best results for deriving leaf area (see Figure 3.10 in the Results).

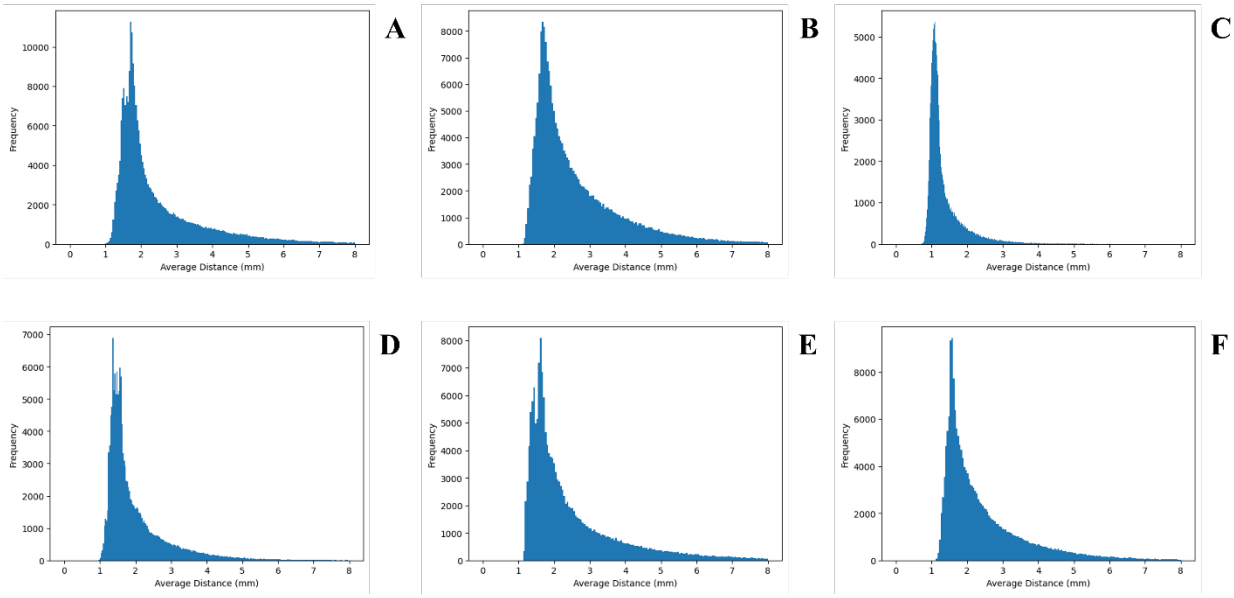


Figure 3.3 Average distance between data points and six nearest neighbors for plants (A) 5N, (B) 6N, (C) 7N, (D) 8N, (E) 13S, (F) 15S.

Once the mesh was created, the leaf area was calculated by summing the area of the triangles in the mesh. Linear regression was performed between the BPA mesh derived leaf area and the measured (ground truth) leaf area.

Biomass was regressed from both the number of voxels in the point cloud and the BPA mesh derived leaf area. Note, the ground truth biomass is fresh biomass, not dry biomass. Voxel sizes of 1 mm, 2 mm, 4 mm, and 8mm were used for voxelization.

3.3.5 Evaluation

Precision, recall, dice (equivalent to F_1), and IoU were used to evaluate the 2D semantic segmentation used in the proposed phenotyping pipeline. These are commonly used metrics that describe both the successes (true positives and true negatives) and failures (false positives and false negatives) of the segmentation as well as overall segmentation accuracy. The definitions of these metrics are:

$$Precision = \frac{TP}{TP + FP} \quad (1)$$

$$Recall = \frac{TP}{TP + FN} \quad (2)$$

$$Dice = \frac{2|G \cap S|}{|G| + |S|} \quad (3)$$

$$IoU = \frac{|G \cap S|}{|G \cup S|} \quad (4)$$

where

TP = Number of correctly identified positive voxels (true positives),

FP = Number of incorrectly identified positive voxels (false positives),

FN = Number of incorrectly identified negative voxels (false negatives),

G = Ground truth segmentation, and

S = Predicted segmentation.

To evaluate the computer vision-assisted (predicted) measurement method, root mean square error (RMSE) and mean absolute percent error (MAPE) were used.

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \quad (5)$$

$$MAPE = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{y_i - \hat{y}_i}{y_i}\right| \times 100\% \quad (6)$$

where

n = Number of samples,

y_i = measured value, and

$\hat{y}_i$ = predicted value.

Bland-Altman analysis (J. M. Bland & Altman, 1986) was used to measure the agreement between the ground truth measurements for vine length and leaf area and the predicted measurements. The difference (Equation 7), bias (Equation 8), standard deviation (SD) (Equation 9), and limits of agreement (LoA) (Equation 10 and Equation 11) were calculated for the Bland-Altman analysis. Additionally, the p-value of the regression line was examined to determine if

the bias was constant over the measurement range. A p-value greater than 0.05 indicates there is not a variation in bias over the measurement range.

$$\text{Difference} = \text{Predicted} - \text{Ground truth} \quad (7)$$

$$\text{Bias} = \text{Mean}(\text{Difference}) \quad (8)$$

$$\text{SD} = \text{Standard Deviation}(\text{Difference}) \quad (9)$$

$$\text{Upper LoA} = \text{Bias} + 1.96 \times \text{SD} \quad (10)$$

$$\text{Lower LoA} = \text{Bias} - 1.96 \times \text{SD} \quad (11)$$

3.3.6 Environment

The experiments were conducted on a laptop (Creator M16, MSI, Taiwan) with an Intel Core i7 CPU at 2.30 GHz, 32 GB of RAM, and an NVIDIA GeForce RTX 3050 GPU. Python (van Rossum, 2022) 3.7.12 (image and point cloud retrieval) and 3.10.10 (segmentation and phenotypic trait measurement) were used for the experiments. Model training was performed using Google Colab Pro+.

3.4 Results

3.4.1 Semantic Segmentation

Table 3.2 presents segmentation accuracy metrics for both semantic segmentation with the fine-tuned SegFormer MiT-b0 model and thresholding Hue Saturation Value (HSV) images by values between (35, 30, 30) and (70, 255, 255). These values were selected by hand tuning.

Please note these scores are averaged for the test set. Although segmentation with SegFormer MiT-b0 outperformed segmentation by thresholding, these results can be misleading. The manual annotations used as ground truth for model training often over segmented the hop plants, causing the segmentation by thresholding to under segment the ground truth annotations. However, training a deep learning model did prevent the segmentation of regions that fell within the determined HSV threshold but did not contain hop plants. Therefore, training a model to segment the images is a crucial step for scenes that contain noisy backgrounds.

Table 3.2 Segmentation accuracy metrics obtained by semantic segmentation with SegFormer MiT-b0 and Thresholding by HSV values.

Method	Precision	Recall	Dice	IoU
Segmentation with SegFormer MiT-b0	0.806	0.766	0.769	0.681
Thresholding by HSV values	0.352	0.250	0.260	0.404

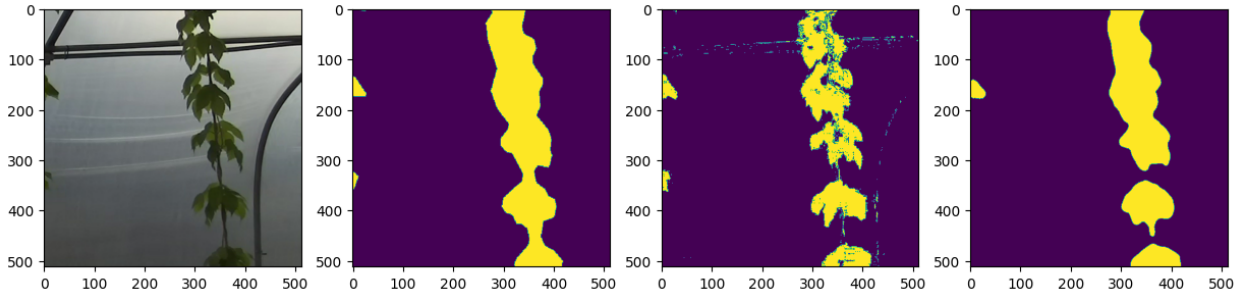


Figure 3.4 Example segmentations: input image (far left), manual segmentation (middle left), segmentation by thresholding (middle right), segmentation by SegFormer MiT-b0 (far right).

3.4.2 Phenotypic Trait Measurement

Figure 3.5 presents a scatter plot and linear regression line for the RANSAC derived vine length and the measured vine length (A) and the Bland-Altman plot for the two measurement methods (B). The linear regression line had an R^2 of 0.73. There was an RMSE of 0.250 m and MAPE of 15.2% with a p-value less than 0.05. The Bland-Altman plot showed the automated pipeline underestimated the vine length, but there was not a variation in bias with increasing vine length.

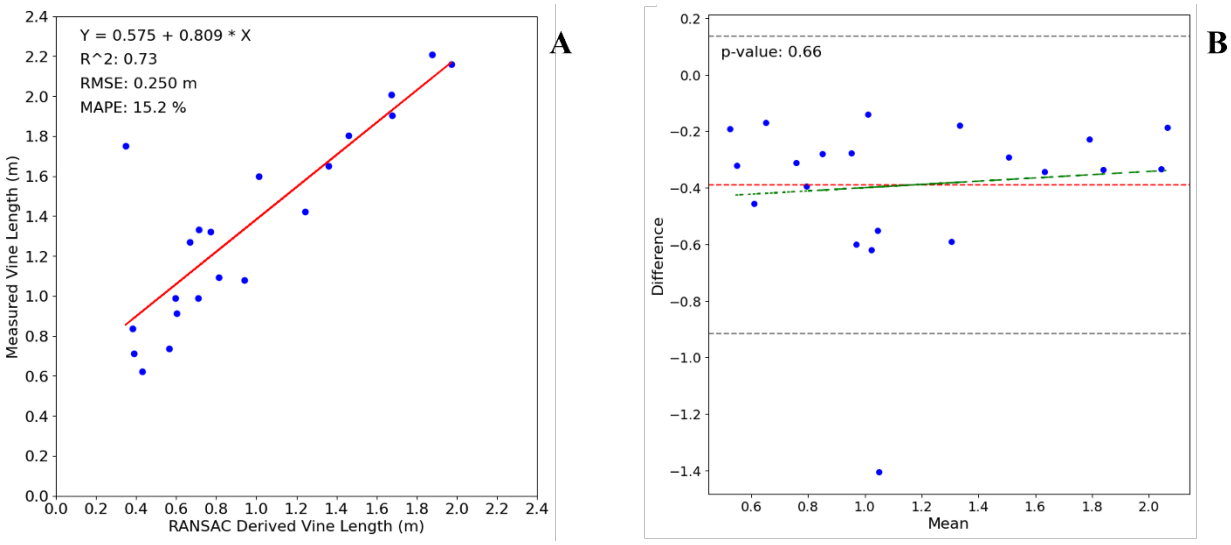


Figure 3.5 Scatter plot of RANSAC derived length and measured length (A) and Bland-Altman plot (B). For B, the bias is represented by a red dashed line and the upper and lower limits of agreement are represented by gray dashed lines. The regression line is shown as a green dashed line.

Leaves could be reliably segmented from the images, however, vine material at the top of the plant was not reliably segmented. Because the ground truth length measurements included this vine material, the RANSAC derived vine length under predicted the ground truth vine length. An example of missing vine is shown in Figure 3.6.

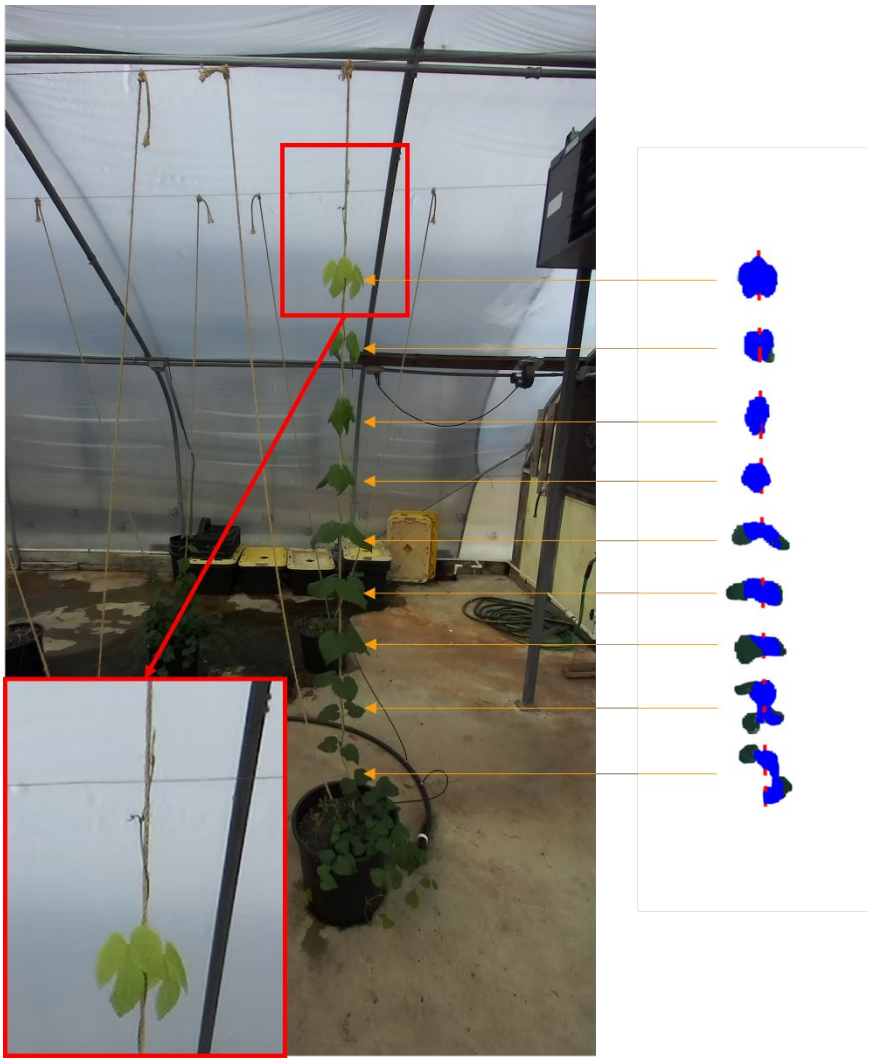


Figure 3.6 Plant 8N imaged on December 09, 2021. Vine missing from segmentation highlighted at bottom left. Meaured and RANSAC derived vine length were 2.210 m and 1.877 m, respectively, for a difference of 0.333 m. Line fitted by RANSAC is in red and inliers are in blue (right). Note, plant material below 240 mm was removed in the segmented point cloud.

The outlier present in Figure 3.5.A represented plant 8N imaged on December 03, 2021 (Figure 3.7). The RANSAC line was fitted to plant material at the bottom of the vine that was not

removed when removing plant material below 340 mm. Therefore, it should be observed that this method is suitable for measuring vine length of plants without excess growth at the base of the plant. However, it is common to remove excess growth that is not being trained upwards because the dense growth creates conditions ideal for mildew diseases (Neve, 1991).

Figure 3.7 Plant 8N imaged on December 03, 2021. Meaured and RANSAC derived vine length were 1.753 m and 0.347 m, respectively, for a difference of 1.406 m. Line fitted by RANSAC is in red and inliers are in blue (right). Note, plant material below 240 mm was removed in the segmented point cloud.

Removing this outlier from the dataset prior to linear regression produced a regression line with an R^2 of 0.91. There was an RMSE of 0.142 m and MAPE of 10.6% with a p-value less than 0.05 (Figure 3.8). The Bland-Altman plot showed the automated pipeline underestimated the vine length, but there was not a variation in bias with increasing vine length.

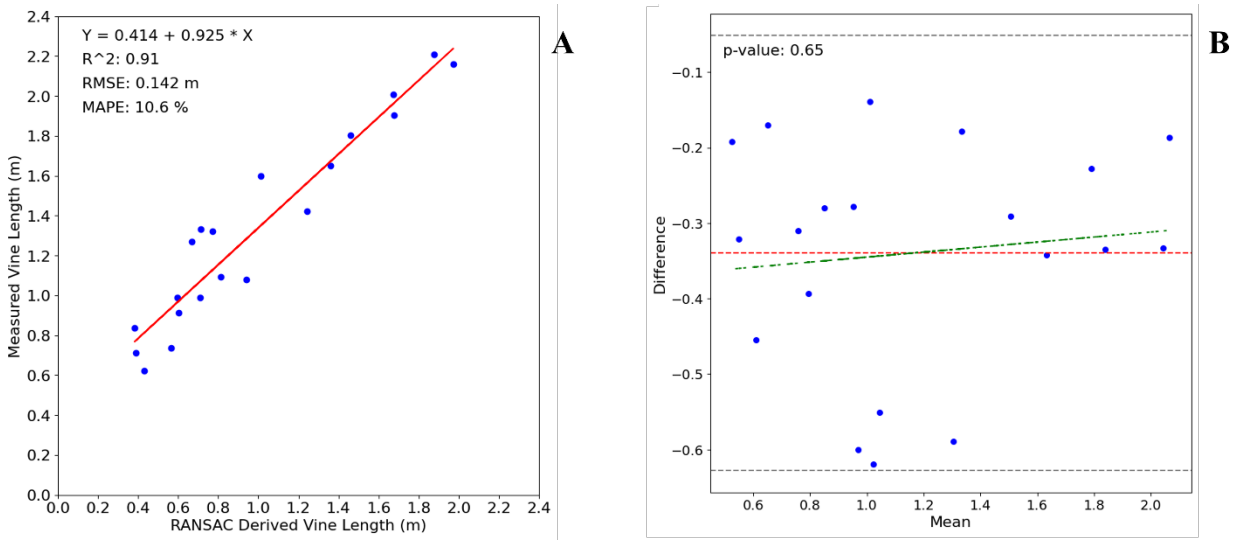


Figure 3.8 Scatter plot of RANSAC derived length and measured length (A) and Bland-Altman plot (B) with plant 8N on December 03, 2021 removed. For B, the bias is represented by a red dashed line and the upper and lower limits of agreement are represented by gray dashed lines. The regression line is shown as a green dashed line.

Adjusting the height under which the pot and extra plant material would be removed to 360 mm to include plant 8N on December 03, 2021, produced a regression line with an R^2 of 0.79. There was an RMSE of 0.219 m and MAPE of 15.2% with a p-value less than 0.05 (Figure 3.9). The Bland-Altman plot showed the automated pipeline underestimated the vine length, but there was not a variation in bias with increasing vine length.

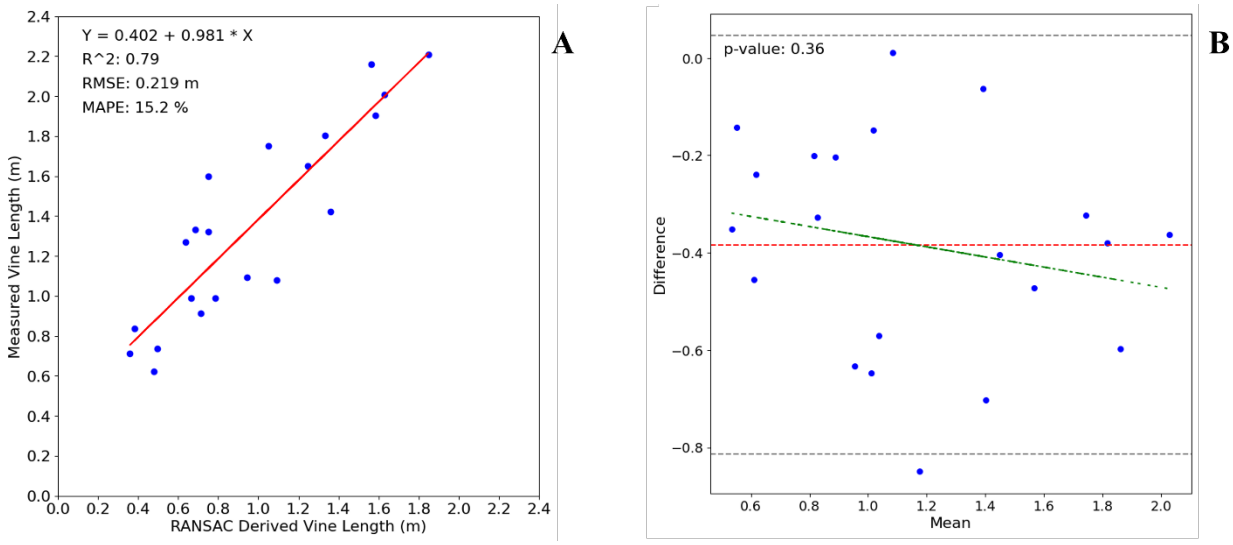


Figure 3.9 Scatter plot of RANSAC derived length and measured length (A) and Bland-Altman plot (B) after adjusting height under which point cloud data would be removed. For B, the bias is represented by a red dashed line and the upper and lower limits of agreement are represented by gray dashed lines. The regression line is shown as a green dashed line.

Figure 3.10 presents scatter plots and linear regression lines for the BPA mesh derived leaf area and the measured leaf area (A-B-C) and the Bland-Altman plots for the two measurement methods (D-E-F). Results are presented for radii of [2, 1] mm (left), [4, 2, 1] mm (middle), and [8, 4, 2, 1] mm (right). The linear regression line for results obtained from radii of [4, 2, 1] mm achieved the highest R^2 of 0.95 and the lowest RMSE and MAPE of 0.050 m^2 and 6.6%, respectively. All p-values were less than 0.05. Except for BPA derived leaf area using radii of [2,1] mm, the Bland-Altman plot showed there was not a variation in bias with increasing leaf area.

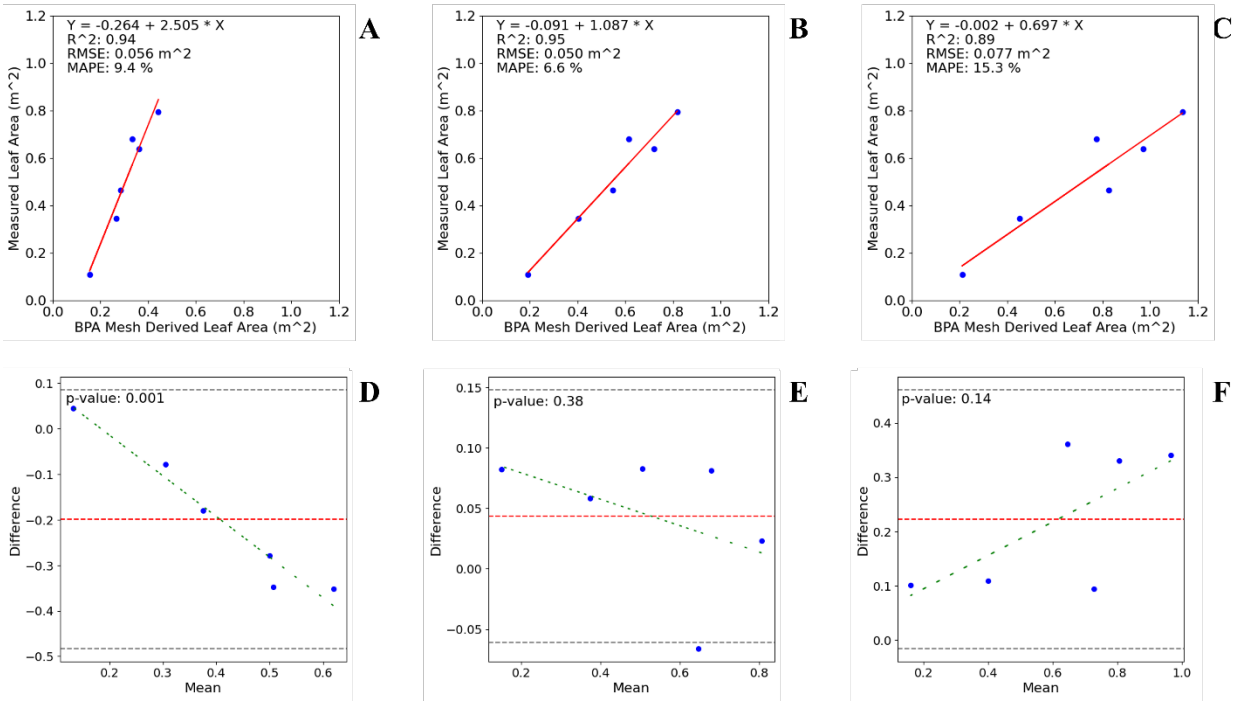


Figure 3.10 Scatter plots of BPA mesh dervied leaf area and measured leaf area (top) and Bland-Altman plots (bottom) for radii of (A-D) [2, 1] mm, (B-E) [4, 2, 1] mm, (C-F) [8, 4, 2, 1] mm. The bias is represented by a red dashed line and the upper and lower limits of agreement are represented by gray dashed lines. The regression line is shown as a green dashed line.

Figure 3.11 presents scatter plots and linear regression lines for the voxel count derived biomass and the measured biomass (A-B-C). Results are presented for voxel sizes of 2 mm (left), 4 mm (middle), and 8 mm (right). The linear regression line for results obtained from a voxel size of 4 mm achieved the highest R^2 of 0.87. All p-values were less than 0.05.

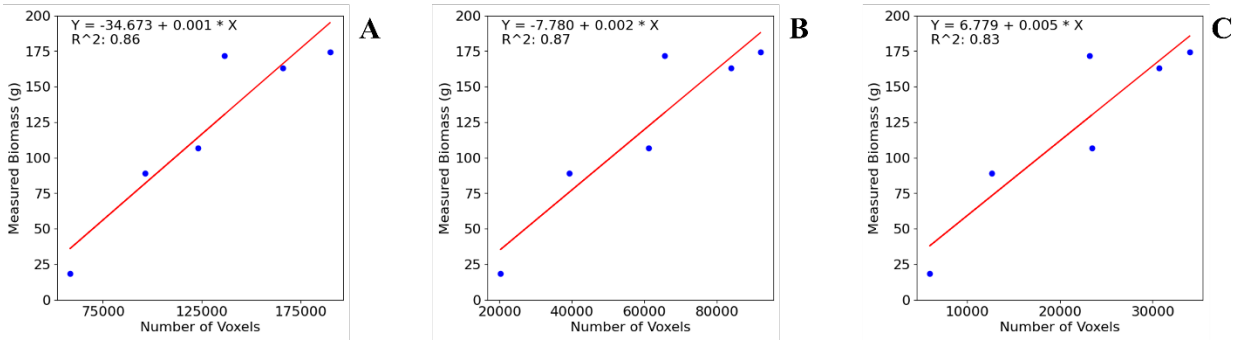


Figure 3.11 Scatter plots of voxel derived biomass and measured biomass (top) for voxel sizes of (A) 2 mm, (B) 4 mm, (C) 8 mm.

Figure 3.12 presents scatter plots and linear regression lines for the leaf area derived biomass and the measured biomass (A-B-C). Results are presented for radii of [2, 1] mm (left), [4, 2, 1] mm (middle), and [8, 4, 2, 1] mm (right). The linear regression line for results obtained from radii of [4, 2, 1] mm achieved the highest R^2 of 0.91. All p-values were less than 0.05. Note, biomass was regressed from the BPA mesh derived leaf area.

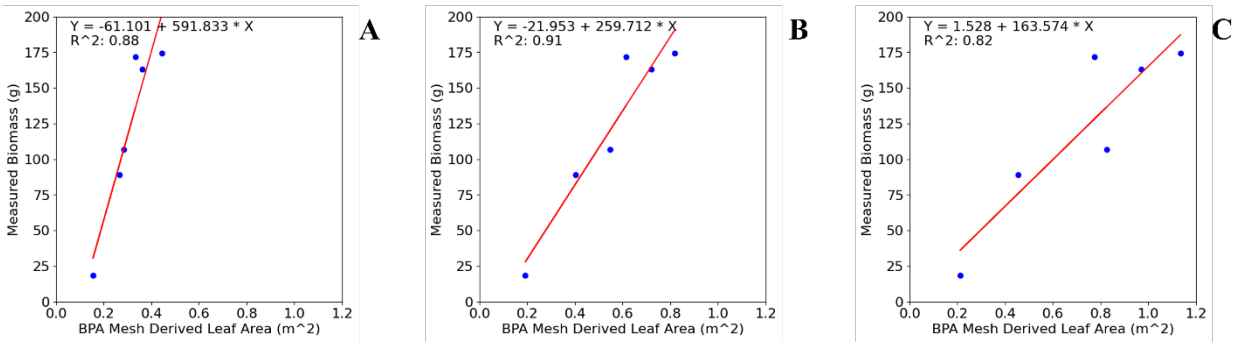


Figure 3.12 Scatter plots of leaf area dervied biomass and measured biomass (top) for radii of (A) [2, 1] mm, (B) [4, 2, 1] mm, (C) [8, 4, 2, 1] mm.

3.5 Discussion

This study presents a method for vine length, leaf area, and biomass measurement from 3D point clouds of hop vines in greenhouse. The proposed method achieved R^2 values of 0.79 (Figure 3.9), 0.95 (Figure 3.10), and 0.91 (Figure 3.12) for vine length, leaf area, and biomass, respectively. Furthermore, Bland-Altman analysis demonstrated there was not a variation in bias with an increase of vine length or leaf area.

For the length measurements obtained by removing point cloud data below a height of 360 mm (Figure 3.9), the vine length was underestimated by 0.384 m on average. One solution to the underestimation problem would be to implement a measurement standard of measuring vine length from the topmost leaf. It is also important to note that adjusting the parameters for line fitting with RANSAC and the height under which point cloud data is removed for each plant would yield better results (Figure 3.5, Figure 3.8, Figure 3.9). However, this is implausible for a high-throughput pipeline. Future work could include developing a method of automated parameter tuning to find ideal parameters for line fitting with RANSAC or to determine the height under which to remove extra plant material not contained in the vine of interest. Alternatively, a method that does not require parameter tuning could be implemented.

We tested different combinations of radii for creating a triangle mesh with BPA for leaf area measurement from 3D point clouds. After examining the histograms of the average distances between each point and its six nearest neighbors and empirically testing radii combinations, we found that radii of [4, 2, 1] mm achieved the highest R^2 value of 0.95, however radii of [2, 1] mm achieved a comparable a comparable R^2 value of 0.94. These results were

corroborated by the nearest neighbor histograms, which often had peaks around 2 mm, but with average distances above 2 mm as well (Figure 3.3). Our result for regression of leaf area is comparable with the results achieved by Klodt et al. in 2015, with an R^2 0.93 for regression of leaf area obtained by weighting segmented leaf pixels according to their depth in a depth map.

We compared linear regression for biomass estimation from both BPA mesh derived leaf area and voxelized point clouds. Regressing biomass from leaf area achieved a greater value for R^2 than regressing biomass from voxel count, with R^2 values of 0.91 and 0.87 respectively using a maximum radius and voxel size of 4 mm (Figure 3.12.B and Figure 3.11.B).

Future work could include automated isolation of the plant of interest from the 3D scene. Second, investigation into solutions solving the underestimation problem from missing vine segmentation could be investigated, including: study of segmentation model architecture, study of increasing resolution in the stereo images, and study of vine length measurement accuracy when measuring vine length from the top leaf. Third, automated parameter tuning for ideal parameters for vine length measurement could be developed.

3.6 Conclusion

This project developed and evaluated a semi-automated pipeline for measuring vine length, leaf area, and biomass of hop plants in greenhouse. We implemented computationally efficient methods for measuring these phenotypic traits and achieved R^2 values of 0.79, 0.95, and 0.91 for vine length, leaf area, and biomass. This pipeline offers a tool for researchers for high-throughput and non-destructive morphological phenotyping for hop plants in variety trials.

3.7 Acknowledgements

This project was supported by the Alabama Agricultural Experiment Station and the Hatch program of the National Institute of Agriculture and Food, U.S. Department of Agriculture.

3.8 References

Almaguer, C., Schönberger, C., Gastl, M., Arendt, E. K., & Becker, T. (2014). Humulus lupulus - a story that begs to be told. A review. *Journal of the Institute of Brewing*, n/a-n/a. https://doi.org/10.1002/jib.160

Artzet, S., Chen, T.-W., Chopard, J., Brichet, N., Mielewczik, M., Cohen-Boulakia, S., Cabrera-Bosquet, L., Tardieu, F., Fournier, C., & Pradal, C. (2019). Phenomenal: An automatic open source library for 3D shoot architecture reconstruction and analysis for image-based plant phenotyping. *BioRxiv*. https://doi.org/10.1101/805739

Azzari, G., Goulden, M., & Rusu, R. (2013). Rapid characterization of vegetation structure with a Microsoft Kinect sensor. *Sensors*, *13*(2), 2384–2398. https://doi.org/10.3390/s130202384

Bernardini, F., Mittleman, J., Rushmeier, H., Silva, C., & Taubin, G. (1999). The ball-pivoting algorithm for surface reconstruction. *IEEE Transactions on Visualization and Computer Graphics*, *5*(4), 349–359. https://doi.org/10.1109/2945.817351

Bland, J. M., & Altman, D. (1986). Statistical methods for assessing agreement between two methods of clinical measurement. *The lancet*, *327*(8476), 307-310.

Bland, J. S., Minich, D., Lerman, R., Darland, G., Lamb, J., Tripp, M., & Grayson, N. (2015). Isohumulones from hops (Humulus lupulus) and their potential role in medical nutrition therapy. *PharmaNutrition*, *3*(2), 46–52. https://doi.org/10.1016/j.phanu.2015.03.001

Castro, P. H. N., Moreira, G. J. P., & Luz, E. J. da S. (2022). An End-to-End Deep Learning System for Hop Classification. *IEEE Latin America Transactions*, *20*(3), 430–442. https://doi.org/10.1109/TLA.2022.9667141

Chaivivatrakul, S., Tang, L., Dailey, M. N., & Nakarmi, A. D. (2014). Automatic morphological trait characterization for corn plants via 3D holographic reconstruction. *Computers and Electronics in Agriculture*, *109*, 109–123. https://doi.org/10.1016/j.compag.2014.09.005

CloudCompare (2022). (Version 2.12.4) [Software]. http://www.cloudcompare.org/

Dwyer, B., Nelson, J., Solawetz, J., & et. al. (2022). *Roboflow* (1.0).

Farhanah, A., & Al Maki, W. F. (2022). Hops Plants Disease Detection using Feature Selection based BPSO-SVM. *2022 9th International Conference on Electrical Engineering, Computer Science and Informatics (EECSI)*, 389–393. https://doi.org/10.23919/EECSI56542.2022.9946620

Fischler, M. A., & Bolles, R. C. (1981). Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, *24*(6), 381-395. https://doi.org/10.1145/358669.358692

Gehan, M. A., Fahlgren, N., Abbasi, A., Berry, J. C., Callen, S. T., Chavez, L., Doust, A. N., Feldman, M. J., Gilbert, K. B., Hodge, J. G., Hoyer, J. S., Lin, A., Liu, S., Lizárraga, C., Lorence, A., Miller, M., Platon, E., Tessman, M., & Sax, T. (2017). PlantCV v2: Image

analysis software for high-throughput plant phenotyping. *PeerJ*, *5*, e4088. https://doi.org/10.7717/peerj.4088

Ghosal, S., Blystone, D., Singh, A. K., Ganapathysubramanian, B., Singh, A., & Sarkar, S. (2018). An explainable deep machine vision framework for plant stress phenotyping. *Proceedings of the National Academy of Sciences*, *115*(18), 4613–4618. https://doi.org/10.1073/pnas.1716999115

Jiang, Y., Li, C., & Paterson, A. H. (2016). High throughput phenotyping of cotton plant height using depth images under field conditions. *Computers and Electronics in Agriculture*, *130*, 57–68. https://doi.org/10.1016/j.compag.2016.09.017

Karabín, M., Hudcová, T., Jelínek, L., & Dostálek, P. (2016). Biologically active compounds from hops and prospects for their use. *Comprehensive Reviews in Food Science and Food Safety*, *15*(3), 542–567. https://doi.org/10.1111/1541-4337.12201

Kerkech, M., Hafiane, A., & Canals, R. (2020). VddNet: Vine disease detection network based on multispectral images and depth map. *Remote Sensing*, *12*(20), 3305. https://doi.org/10.3390/rs12203305

Kerkech, M., Hafiane, A., & Canals, R. (2020). Vine disease detection in UAV multispectral images using optimized image registration and deep learning segmentation approach. *Computers and Electronics in Agriculture*, *174*, 105446. https://doi.org/10.1016/j.compag.2020.105446

Klodt, M., Herzog, K., Töpfer, R., & Cremers, D. (2015). Field phenotyping of grapevine growth using dense stereo reconstruction. *BMC Bioinformatics*, *16*(1), 143. https://doi.org/10.1186/s12859-015-0560-x

Knudson, W., Sirrine, J. R., & Mann, J. T. (2020). The Geographic Dispersion of Hop Production in the United States: Back to the Future? In *The Geography of Beer* (pp. 113–120). Springer International Publishing. https://doi.org/10.1007/978-3-030-41654-6_9

Legun, K., Comi, M., & Vicol, M. (2022). New aesthetic regimes: The shifting global political ecology of aroma hops. *Geoforum*, *128*, 148–157. https://doi.org/10.1016/j.geoforum.2021.12.004

Liu, S., Cossell, S., Tang, J., Dunn, G., & Whitty, M. (2017). A computer vision system for early stage grape yield estimation based on shoot detection. *Computers and Electronics in Agriculture*, *137*, 88–101. https://doi.org/10.1016/j.compag.2017.03.013

Mariga, L. (2022). *pyRANSAC-3D*. https://doi.org/10.5281/zenodo.7212567

National Agricultural Statistics Service. (2022, December 19). *2022 USDA-NASS National Hop Report*. https://www.usahops.org/news/2022-usda-nass-national-hop-report

Nellithimaru, A. K., & Kantor, G. A. (2019). ROLS : Robust Object-Level SLAM for Grape Counting. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, 0–0.

Neve, R. A. (1991). *Hops*. Springer Netherlands. https://doi.org/10.1007/978-94-011-3106-3

Niknejad, N., Bidese-Puhl, R., Bao, Y., Payn, K. G., & Zheng, J. (2023). Phenotyping of architecture traits of loblolly pine trees using stereo machine vision and deep learning: Stem diameter, branch angle, and branch diameter. *Computers and Electronics in Agriculture*, *211*, 107999. https://doi.org/10.1016/j.compag.2023.107999

Ninomiya, S. (2022). High-throughput field crop phenotyping: current status and challenges. *Breeding Science*, *72*(1), 21069. https://doi.org/10.1270/jsbbs.21069

Pothen, Z. S., & Nuske, S. (2016). Texture-based fruit detection via images using the smooth patterns on the fruit. *2016 IEEE International Conference on Robotics and Automation (ICRA)*, 5171–5176. https://doi.org/10.1109/ICRA.2016.7487722

Ramos-Giraldo, P., Reberg-Horton, C., Locke, A. M., Mirsky, S., & Lobaton, E. (2020). Drought stress detection using low-cost computer vision systems and machine learning techniques. *IT Professional*, *22*(3), 27–29. https://doi.org/10.1109/MITP.2020.2986103

Roy, A. M., & Bhaduri, J. (2021). A deep learning enabled multi-class plant disease detection model based on computer vision. *AI*, *2*(3), 413–428. https://doi.org/10.3390/ai2030026

Schönberger, C., & Kostelecky, T. (2011). 125th anniversary review: The role of hops in brewing. *Journal of the Institute of Brewing*, *117*(3), 259–267. https://doi.org/10.1002/j.2050-0416.2011.tb00471.x

Singh, V., Sharma, N., & Singh, S. (2020). A review of imaging techniques for plant disease detection. *Artificial Intelligence in Agriculture*, *4*, 229–242. https://doi.org/10.1016/j.aiia.2020.10.002

Stereolabs. ZED SDK. https://www.stereolabs.com

Turner, S. F., Benedict, C. A., Darby, H., Hoagland, L. A., Simonson, P., Sirrine, J. R., & Murphy, K. M. (2011). Challenges and opportunities for organic hop production in the United States. *Agronomy Journal*, *103*(6), 1645–1654. https://doi.org/10.2134/agronj2011.0131

van Rossum, G. (2022). *Python 3 Reference Manual*. https://docs.python.org/release/3.10.9/

Wang, Y., Chen, Y., Wang, J., Chen, J., Aggarwal, B. B., Pang, X., & Liu, M. (2012). Xanthohumol, a prenylated chalcone derived from hops, suppresses cancer cell invasion through inhibiting the expression of CXCR4 chemokine receptor. *Current Molecular Medicine*, *12*(2), 153–162. https://doi.org/10.2174/156652412798889072

White, J. W., Andrade-Sanchez, P., Gore, M. A., Bronson, K. F., Coffelt, T. A., Conley, M. M., Feldmann, K. A., French, A. N., Heun, J. T., Hunsaker, D. J., Jenks, M. A., Kimball, B. A., Roth, R. L., Strand, R. J., Thorp, K. R., Wall, G. W., & Wang, G. (2012). Field-based phenomics for plant genetics research. *Field Crops Research*, *133*, 101–112. https://doi.org/10.1016/j.fcr.2012.04.003

Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., Cistac, P., Rault, T., Louf, R., Funtowicz, M., Davison, J., Shleifer, S., von Platen, P., Ma, C., Jernite, Y., Plu, J., Xu, C., Le Scao, T., Gugger, S., … Rush, A. (2020). Transformers: State-of-the-Art Natural Language Processing. *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, 38–45. https://doi.org/10.18653/v1/2020.emnlp-demos.6

Xiang, L., Bao, Y., Tang, L., Ortiz, D., & Salas-Fernandez, M. G. (2019). Automated morphological traits extraction for sorghum plants via 3D point cloud data analysis. *Computers and Electronics in Agriculture*, *162*, 951–961. https://doi.org/10.1016/j.compag.2019.05.043

Xie, E., Wang, W., Yu, Z., Anandkumar, A., Alvarez, J. M., & Luo, P. (2021). SegFormer: Simple and efficient design for semantic segmentation with transformers. *Advances in Neural Information Processing Systems*, *34*, 12077-12090. https://doi.org/10.48550/arXiv.2105.15203

Zanoli, P., Rivasi, M., Zavatti, M., Brusiani, F., & Baraldi, M. (2005). New insight in the neuropharmacological activity of Humulus lupulus L. *Journal of Ethnopharmacology*, *102*(1), 102–106. https://doi.org/10.1016/j.jep.2005.05. 040

Zhou, Q.-Y., Park, J., & Koltun, V. (2018). *Open3D: A Modern Library for 3D Data Processing*. http://www.open3d.org/

Supplementary Materials

Table S1. Complete segmentation accuracy metrics obtained by two-dimensional instance segmentation for the complete scans.

Training View(s)	Prediction View(s)	Plant	Precision	Recall	Dice	IoU
Z	Z	Poinsettia1	0.872	0.921	0.896	0.812
Z	Z	Onion1	0.910	0.926	0.918	0.848
Z	Z	Poinsettia2	0.900	0.976	0.937	0.881
Z	Z	Onion2	0.817	0.982	0.892	0.805
Average			0.875	0.951	0.911	0.836
XYZ	X	Poinsettia1	0.830	0.732	0.778	0.637
XYZ	X	Onion1	0.885	0.177	0.295	0.173
XYZ	X	Poinsettia2	0.899	0.952	0.925	0.860
XYZ	X	Onion2	0.829	0.943	0.882	0.789
Average			0.861	0.701	0.720	0.615
XYZ	Y	Poinsettia1	0.837	0.798	0.817	0.691
XYZ	Y	Onion1	0.887	0.196	0.321	0.191
XYZ	Y	Poinsettia2	0.889	0.951	0.919	0.851
XYZ	Y	Onion2	0.835	0.952	0.890	0.801
Average			0.862	0.725	0.737	0.634
XYZ	Z	Poinsettia1	0.882	0.895	0.888	0.799
XYZ	Z	Onion1	0.508	0.600	0.550	0.380

XYZ	Z	Poinsettia2	0.919	0.969	0.943	0.893
XYZ	Z	Onion2	0.853	0.957	0.902	0.822
Average			0.791	0.855	0.821	0.723
XYZ	XY	Poinsettia1	0.770	0.888	0.825	0.702
XYZ	XY	Onion1	0.885	0.310	0.459	0.298
XYZ	XY	Poinsettia2	0.848	0.976	0.908	0.831
XYZ	XY	Onion2	0.769	0.975	0.860	0.754
Average			0.818	0.787	0.763	0.646
XYZ	XZ	Poinsettia1	0.797	0.926	0.857	0.750
XYZ	XZ	Onion1	0.514	0.630	0.566	0.395
XYZ	XZ	Poinsettia2	0.870	0.984	0.924	0.858
XYZ	XZ	Onion2	0.785	0.983	0.873	0.774
Average			0.742	0.881	0.805	0.694
XYZ	YZ	Poinsettia1	0.801	0.917	0.855	0.747
XYZ	YZ	Onion1	0.518	0.639	0.572	0.400
XYZ	YZ	Poinsettia2	0.866	0.981	0.920	0.852
XYZ	YZ	Onion2	0.799	0.982	0.881	0.788
Average			0.746	0.880	0.807	0.697
XYZ	XYZ	Poinsettia1	0.741	0.938	0.828	0.707
XYZ	XYZ	Onion1	0.522	0.662	0.584	0.412
XYZ	XYZ	Poinsettia2	0.834	0.988	0.905	0.826
XYZ	XYZ	Onion2	0.750	0.987	0.853	0.743
Average			0.712	0.894	0.792	0.672

Table S2. Complete segmentation accuracy metrics obtained by two-dimensional instance segmentation for the truncated scans.

Training View(s)	Prediction View(s)	Plant	Precision	Recall	Dice	IoU
Z	Z	Poinsettia1	0.821	0.866	0.843	0.729
Z	Z	Onion1	0.533	0.877	0.663	0.496
Z	Z	Poinsettia2	0.808	0.956	0.876	0.779
Z	Z	Onion2	0.599	0.968	0.740	0.587
Average			0.690	0.917	0.781	0.648
XYZ	X	Poinsettia1	0.826	0.572	0.676	0.511
XYZ	X	Onion1	0.755	0.768	0.762	0.615
XYZ	X	Poinsettia2	0.827	0.926	0.874	0.776
XYZ	X	Onion2	0.620	0.901	0.734	0.580
Average			0.757	0.792	0.761	0.620
XYZ	Y	Poinsettia1	0.845	0.705	0.769	0.624
XYZ	Y	Onion1	0.671	0.614	0.641	0.472
XYZ	Y	Poinsettia2	0.798	0.935	0.861	0.757
XYZ	Y	Onion2	0.632	0.910	0.745	0.594
Average			0.736	0.791	0.754	0.612
XYZ	Z	Poinsettia1	0.806	0.833	0.819	0.694
XYZ	Z	Onion1	0.778	0.827	0.802	0.669
XYZ	Z	Poinsettia2	0.857	0.941	0.897	0.814
XYZ	Z	Onion2	0.674	0.947	0.787	0.649
Average			0.779	0.887	0.826	0.706
XYZ	XY	Poinsettia1	0.786	0.812	0.799	0.665

XYZ	XY	Onion1	0.648	0.865	0.741	0.588
XYZ	XY	Poinsettia2	0.736	0.964	0.835	0.716
XYZ	XY	Onion2	0.536	0.961	0.688	0.524
Average			0.676	0.901	0.766	0.624
XYZ	XZ	Poinsettia1	0.753	0.872	0.808	0.678
XYZ	XZ	Onion1	0.680	0.883	0.768	0.624
XYZ	XZ	Poinsettia2	0.776	0.966	0.861	0.756
XYZ	XZ	Onion2	0.559	0.971	0.709	0.550
Average			0.692	0.923	0.787	0.652
XYZ	YZ	Poinsettia1	0.758	0.859	0.806	0.675
XYZ	YZ	Onion1	0.668	0.868	0.755	0.606
XYZ	YZ	Poinsettia2	0.763	0.959	0.850	0.739
XYZ	YZ	Onion2	0.583	0.970	0.729	0.573
Average			0.693	0.914	0.785	0.648
XYZ	XYZ	Poinsettia1	0.716	0.888	0.793	0.657
XYZ	XYZ	Onion1	0.613	0.904	0.731	0.576
XYZ	XYZ	Poinsettia2	0.713	0.973	0.823	0.699
XYZ	XYZ	Onion2	0.512	0.977	0.672	0.506
Average			0.639	0.936	0.755	0.610

Table S3. Complete segmentation accuracy metrics obtained by two-dimensional instance segmentation and noise removal with DBSCAN for the complete scans.

Training View(s)	**Prediction View(s)**	**Plant**	**Precision**	**Recall**	**Dice**	**IoU**
Z	Z	Poinsettia1	0.870	0.912	0.891	0.803

Z	Z	Onion1	0.952	0.919	0.935	0.879
Z	Z	Poinsettia2	0.913	0.975	0.943	0.892
Z	Z	Onion2	0.891	0.972	0.929	0.868
Average			0.906	0.945	0.925	0.860
XYZ	X	Poinsettia1	0.833	0.735	0.781	0.641
XYZ	X	Onion1	0.931	0.171	0.288	0.169
XYZ	X	Poinsettia2	0.904	0.954	0.928	0.866
XYZ	X	Onion2	0.885	0.940	0.912	0.838
Average			.889	0.700	0.728	0.629
XYZ	Y	Poinsettia1	0.834	0.790	0.812	0.683
XYZ	Y	Onion1	0.930	0.200	0.330	0.197
XYZ	Y	Poinsettia2	0.894	0.953	0.923	0.856
XYZ	Y	Onion2	0.905	0.950	0.927	0.864
Average			0.891	0.723	0.748	0.650
XYZ	Z	Poinsettia1	0.875	0.887	0.881	0.787
XYZ	Z	Onion1	0.467	0.590	0.521	0.353
XYZ	Z	Poinsettia2	0.925	0.968	0.946	0.898
XYZ	Z	Onion2	0.927	0.934	0.931	0.870
Average			0.798	0.845	0.820	0.727
XYZ	XY	Poinsettia1	0.768	0.891	0.825	0.702
XYZ	XY	Onion1	0.922	0.326	0.482	0.317
XYZ	XY	Poinsettia2	0.855	0.978	0.912	0.839
XYZ	XY	Onion2	0.813	0.976	0.887	0.798
Average			0.840	0.793	0.777	0.664
XYZ	XZ	Poinsettia1	0.791	0.921	0.851	0.741

XYZ	XZ	Onion1	0.490	0.639	0.554	0.383
XYZ	XZ	Poinsettia2	0.880	0.982	0.928	0.866
XYZ	XZ	Onion2	0.831	0.978	0.898	0.816
Average			0.748	0.880	0.808	0.701
XYZ	YZ	Poinsettia1	0.792	0.912	0.848	0.735
XYZ	YZ	Onion1	0.507	0.643	0.567	0.396
XYZ	YZ	Poinsettia2	0.875	0.979	0.924	0.859
XYZ	YZ	Onion2	0.849	0.977	0.909	0.832
Average			0.756	0.878	0.812	0.706
XYZ	XYZ	Poinsettia1	0.733	0.933	0.821	0.697
XYZ	XYZ	Onion1	0.500	0.676	0.575	0.404
XYZ	XYZ	Poinsettia2	0.845	0.986	0.910	0.835
XYZ	XYZ	Onion2	0.793	0.983	0.878	0.783
Average			0.718	0.895	0.796	0.680

Table S4. Complete segmentation accuracy metrics obtained by two-dimensional instance segmentation and noise removal with DBSCAN for the truncated scans.

Training View(s)	**Prediction View(s)**	**Plant**	**Precision**	**Recall**	**Dice**	**IoU**
Z	Z	Poinsettia1	0.813	0.835	0.824	0.700
Z	Z	Onion1	0.559	0.799	0.658	0.490
Z	Z	Poinsettia2	0.849	0.950	0.897	0.812
Z	Z	Onion2	0.723	0.945	0.819	0.694
Average			0.736	0.882	0.799	0.674
XYZ	X	Poinsettia1	0.833	0.536	0.652	0.484

XYZ	X	Onion1	0.819	0.704	0.757	0.609
XYZ	X	Poinsettia2	0.845	0.918	0.880	0.786
XYZ	X	Onion2	0.727	0.866	0.791	0.654
Average			0.806	0.756	0.770	0.633
XYZ	Y	Poinsettia1	0.849	0.630	0.723	0.566
XYZ	Y	Onion1	0.713	0.524	0.604	0.432
XYZ	Y	Poinsettia2	0.812	0.920	0.863	0.759
XYZ	Y	Onion2	0.774	0.882	0.825	0.702
Average			0.787	0.739	0.754	0.615
XYZ	Z	Poinsettia1	0.787	0.804	0.796	0.661
XYZ	Z	Onion1	0.838	0.743	0.788	0.649
XYZ	Z	Poinsettia2	0.881	0.927	0.904	0.824
XYZ	Z	Onion2	0.816	0.883	0.848	0.736
Average			0.831	0.839	0.834	0.718
XYZ	XY	Poinsettia1	0.793	0.812	0.803	0.670
XYZ	XY	Onion1	0.686	0.848	0.758	0.611
XYZ	XY	Poinsettia2	0.759	0.957	0.847	0.734
XYZ	XY	Onion2	0.591	0.955	0.730	0.575
Average			0.707	0.893	0.784	0.648
XYZ	XZ	Poinsettia1	0.745	0.864	0.800	0.667
XYZ	XZ	Onion1	0.727	0.873	0.793	0.657
XYZ	XZ	Poinsettia2	0.808	0.960	0.878	0.782
XYZ	XZ	Onion2	0.634	0.956	0.763	0.616
Average			0.729	0.913	0.808	0.681
XYZ	YZ	Poinsettia1	0.747	0.849	0.795	0.659

XYZ	YZ	Onion1	0.699	0.850	0.767	0.622
XYZ	YZ	Poinsettia2	0.791	0.945	0.861	0.757
XYZ	YZ	Onion2	0.669	0.952	0.786	0.647
Average			0.727	0.899	0.802	0.671
XYZ	XYZ	Poinsettia1	0.707	0.880	0.785	0.645
XYZ	XYZ	Onion1	0.654	0.893	0.755	0.606
XYZ	XYZ	Poinsettia2	0.749	0.967	0.844	0.730
XYZ	XYZ	Onion2	0.574	0.965	0.720	0.562
Average			0.671	0.926	0.776	0.636