CSS844 Hollender Sub-Group Code Instructions

Members: Joshua Kaste, Miles Roberts, Thilani Jayakody, Andriana Manousidaki, Kenia Segura Aba, Elyse Vischulis, Jiaxin Yang, Jamell Dacon, Rie Sadohara

Dependencies and Installation Instruction:

System requirements:

- Must have Windows PC for 3D LiDAR image analysis with plantscan3D

Anaconda3:

- Comes with a recent Python installation and is required for all code to run.
- Can be installed by downloading the installer here and following instructions: https://www.anaconda.com/products/individual

Jupyter Notebook:

- Comes installed with Anaconda3
- Can be launched from the start menu
- Once in the Jupyter browser, navigate to any provided .ipynb files and open them. Use Ctrl + Enter to execute cells of code.

PlantScan3D

- Necessary for skeletonization of *Arabidopsis* LiDAR images
- Can be installed by following the instructions here after installing Anaconda3:
- https://plantscan3d.readthedocs.io/en/latest/userguide/installation.html

PlantCV

- Necessary for skeletonization of 2D *Arabidopsis* images
- Can be installed by typing the following into the anaconda3 prompt and pressing ‘y’ as necessary to confirm installation:
    - pip install plantcv

Intel Realsense Viewer

- Necessary for real time visualization of Intel Realsense LiDAR output

Overview of Pipelines:

3D LiDAR Image Analysis:

Image acquisition:

The following section describes the experimental design for acquiring 3D point clouds for *Arabidopsis* using the Intel Realsense L515 LiDAR camera, although further optimizations are still necessary to capture accurate and reproducible alignments. View final report document for details for considerations when taking LiDAR images

- We worked in a room without any windows/ambient light or other objects in the background.
- Plants that could stand up without stakes image the best.
- Under the depth visualization controls, change the min and max distance to fit the space of the room you are working in for easier viewing.
- We observed that objects behind the plant (such as a box) would skew the depth measurements. Even objects off to the side seemed to decrease the resolution of the plant, so we removed everything from the background, although this method needs to be reconsidered to include a reference object (see optimization #2).
- We angled the camera to capture just the vegetative growth of the plant and not the pot.
- The plant was positioned 34.6 cm away from the camera
- The following camera settings were changed from default in the Intel Realsense viewer

Controls	
laser power	0.000
min distance	10.00
receiver gain	18.000
resolution	XGA (1024x768)

Segmentation and export to plantscan3D:

1. Open the 20210414_JK_Angle_Estimation_V3.ipynb notebook in Jupyter
2. Follow the instructions in the Jupyter notebook to load in your .ply file, segment the Arabidopsis points in the point cloud, and export as a .asc file.

Skeletonization in plantscan3D:

1. In the Anaconda3 prompt, type **plantscan3d** to start plantscan3D
2. Once plantscan3D has loaded, go to the File dropdown and click "Import points". Navigate to the .asc file you exported from the Jupyter notebook and load it in.

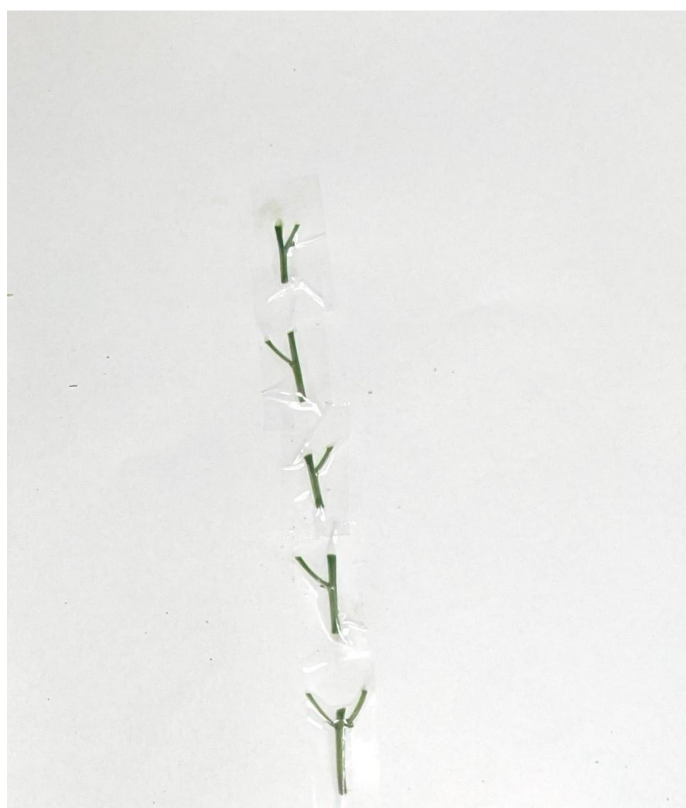
3. Identify the base of the plant. In the Reconstruction dropdown, select “Add Root” and select any option. A node will have been created that you can click and drag. Drag it to the approximate location of the base of the Arabidopsis plant. Rotate around the point cloud to make sure you have correctly placed it, as it can be impossible to tell from a single projection whether you’ve placed it right in 3D space.
4. Once you have placed your root node, in the Reconstruction dropdown again, go to the “Skeletonization” menu and select the Xu et al method. In the screen that pops up, change the bin value to 10 and run the algorithm.
5. The resulting skeletonization will probably not be perfect, but you can manually adjust it.
    a. Remove nodes at the base of the plant that are spuriously created because of the presence of soil surface points
    b. Correct errors in the architecture of the plant by moving nodes around and “re parenting” nodes by right clicking them and selecting that option.
6. Once satisfied with the skeleton, go to File -> Export MTG -> As Node List and export your list of nodes as a .txt file

Branch angle estimation:

1. Once you have your node list as a .txt file, go back to the 20210414_JK_Angle_Estimation_V3.ipynb notebook and start from the heading “Import skeleton from plantscan3d as .txt file (MTG)”. Follow instructions and run code until you get your calculated branch angles.

2D Branch Angle Estimation:

1. Scan or take a picture of *Arabidopsis* branches with the basal part of the segment oriented towards the bottom of the image and the individual branches themselves oriented from top-to-bottom on a *white paper background* (save as a .jpg). If taking a picture, make sure not to include anything outside of the paper background or to crop the image to exclude stuff outside the paper background beforehand. See below for example:



2. In Jupyter, open up the *20210419_JK_Automated_2D_Branch_Angle_Estimator.ipynb* file and run all cells up until 'Code to Run for Angle Estimation'
3. Change the filename in the cell below that heading to whatever you saved your image under
4. Run the cell and in the output it will generate a list of branch angles along with the segmented branch images in the same order. Note that branch angles and images are outputted from top-to-bottom.