

Individual Lab Report #11

Hillel Hochsztein

Wholesome Robotics (Team E)

Teammates: Aman Agarwal, Dung Han Lee, John MacDonald, Aaditya Saraiya

November, 19 2019

Individual Progress

In the past week I have continued worked on the visualizer tool for the monitoring pipeline. A key part of our deliverables is a tool for the farmers to understand the data that we collect and generate. For that reason, I have been focusing on integrating the visualizer into a completed monitoring pipeline. In the week specifically, we completely integrated (and debugged) the entire monitoring pipeline (bag reading, inferencing, and visualization) to run from a single command, reliably. We met as a group to tie up incompatibilities and work on the bash script that launches the individual python scripts in order (and handles file management). Interestingly, some of the python scripts must be run in python2 with ros sourcing, and others must be run in python3 without sourcing ros. To accomplish this we had to come up with a creative series of gnome-terminals to run each script seperately, while waiting for each gnome-terminal to finish (a notoriously difficult caveat) before progressing to the next step.

I also updated the visualizer to handle both weeding and fungus as separate inputs, to be identified as separate items in the visualizer. First, I selected svg icons and used inkscape to convert them to paths, compatible with matplotlib's drawing tools. I picked one icon for pest, one for disease, and a combination of both to signify the presence of both. I then changed the visualizer tool to draw different with different markers (these icons) depending on the present issues, and updated the gui for an individual cluster to allow for editing the fungus values and pest values separately. I also decided to leave color as a scale from red to green, where red means every image in the cluster has both fungus and pest presence, and green means it has no problems whatsoever.

I also researched how to scale the drawing so that it is drawn the same on different computers. By default, matplotlib uses on the computer screen's dpi, to establish the scaling for the markers, but uses the limits of the drawing axes to scale the overall image so the appearance of the visualization looked very different on my computer (with relatively high dpi) than it did on other computers. To solve this I first had to research matplotlib's various backends, as they are numerous and handle scaling differently. The resulting visualization is much easier to read, and allows for the full functionality requested by the farmers (see figures 1 and 2).

Finally, I solved a networking issue we were facing for the robot. Specifically, we have been having issues both with the number of ethernet ports available on the robot, and also the identities of those ports (there are multiple subnets, and the sensor packages need to be attached to exactly one of the subnets. Instead of changing the networking of the robot, I installed a network switch which extends any

one port into 7 ports. I plugged this in to an ethernet port which we have confirmed is on the correct subnet and ordered and installed a buck regulator to power the switch off of the robot's unregulated battery (see figure 3).

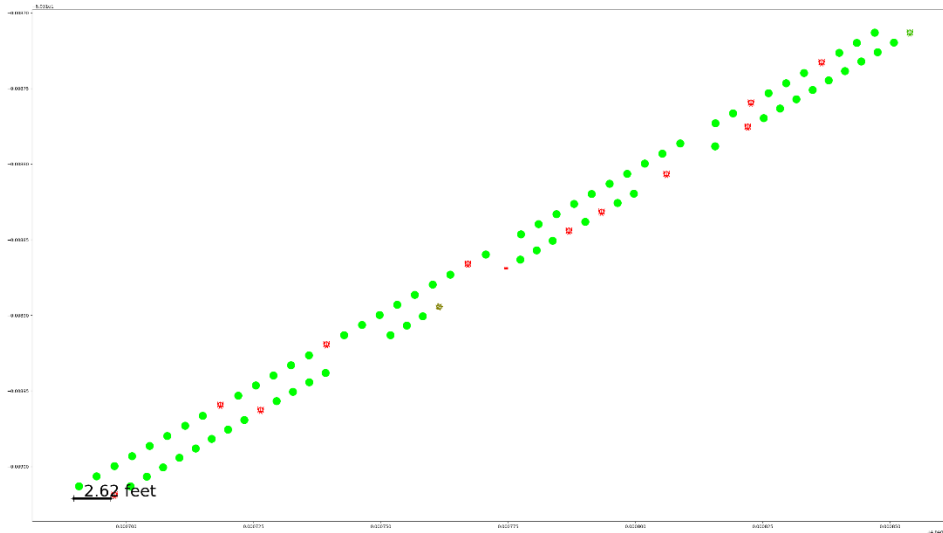


Figure 1 Multiple rows of plants, run through the integrated pipeline, with color indicating severity, and icons indicating type of ailment.

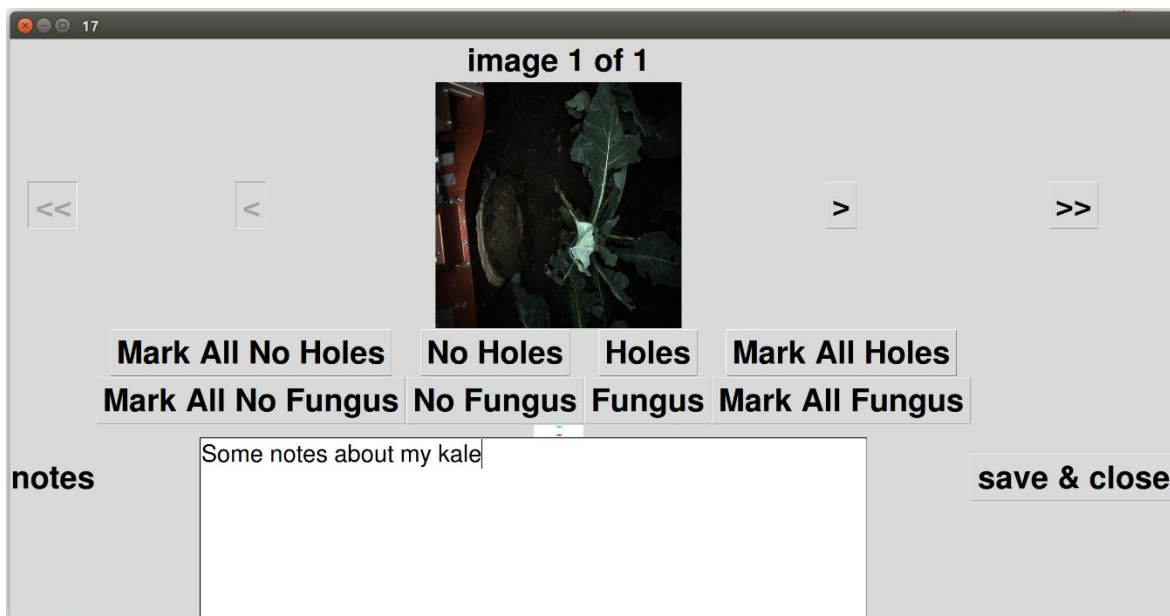


Figure 2 GUI includes readout for image number within the cluster, total cluster status bar, navigation buttons that deactivate when unavailable (i.e. at the first or last image), notepad and Recently added differentiation in controls for fungus and holes.

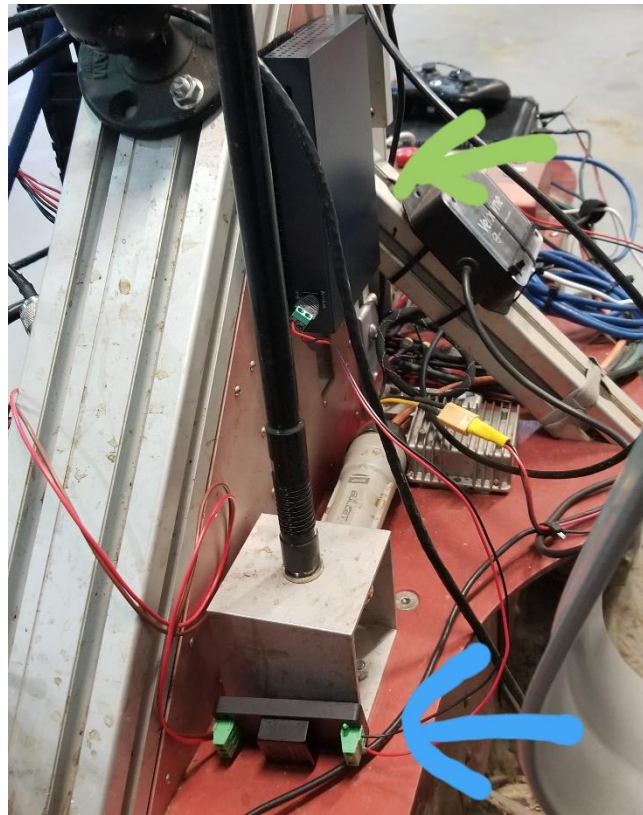


Figure 3 Installation of Network Switch (green arrow) and Power Adapter (blue arrow), affixed with Velcro.

Team Progress

We have been working towards delivery of our MVPs both for monitoring and for navigation. For visualization we are done with our pipeline. We plan to continue adding features and managing bugs in the next few weeks. For navigation we have worked on fusing different row detection and row navigation techniques (such as the lidar based detection, RTK guidance, and visual odometry) to fix issues we found with our row navigation.

Challenges

The biggest challenge I have faced this week has been time management. In addition to almost 20 hours spent on field testing, other classes are beginning to pick up in time demands.

Integration has also posed a minor challenge, just because of stylistic and compatibility issues between my work and that of my teammates. We have successfully managed this by calling meetings as necessary to handle integration. For example, that is how we finished the monitoring pipeline.

Teamwork

Aman has worked on EKF navigation.

Aaditya has been working on visualizer integration and image exposure acceptance for the crop facing camera.

Dung Han has been working on evaluating the plant health model and fine tuning it, including coordinating outsourced data labeling.

John has been working on the particle filter for navigation.

Future Plans

Wholesome Robotics

In the coming week I will be working on tightening up our pipeline integration, to improve reliability. Specifically, I will be looking at a way to handle bag files that are more than one row long, since the current clustering technique is designed for a single row. Finally, I will be available to the rest of the team for help on their tasks, as our time diminishes.