Individual Lab Report

Name:	Dung-Han Lee
Team Name:	Team E Wholesome Robotics
Team members:	Aaditya, Aman, John, Hillel
ILR Number:	07

November 5, 2019

1 individual progress overview

There were one primary focus area in the past 2 weeks, which is switching metric from area ratio to discrete severity level in evaluating Mask-RCNN's performance

1.1 Evaluate Performance for 4 types of plants

1.1.1 Motivation

Since there's a great variety of plants on the field, and relatively limited amount of data on the field on the order of a few hundreds per type of plant), a separate model is trained for each type of 4 plants (Fig 1). And it's desirable to get performance evaluation on these 4 types of plants.

1.1.2 Formulating Features

Recall that, a Mask-RCNN neural network is applied to detect hole and fungus for the monitoring pipeline (Fig 2). The problematic symptoms are segmented out by the model, and their pixel area values e.g. there are 200 hole pixels on a 512 x 512 plant image, are known to the pipeline.

Also recall that, since the data was obtained from a stereo camera, the leave area can also be estimated from a stereo-inferred depth map: anything that's x meter away from the camera are rendered as background, otherwise leave area.

To conduct binary classification, a threshold function (f: if x ¿ threshold, y = 1 ; otherwise y = 0) is applied. Two features are considered as input feature for this progress review (1) the sum of pixel area (2) the ratio of symptom pixel area to the leave area.

(1) is easier to implement while (2) is expected to be more robust to plant growth and the variant in camera height.

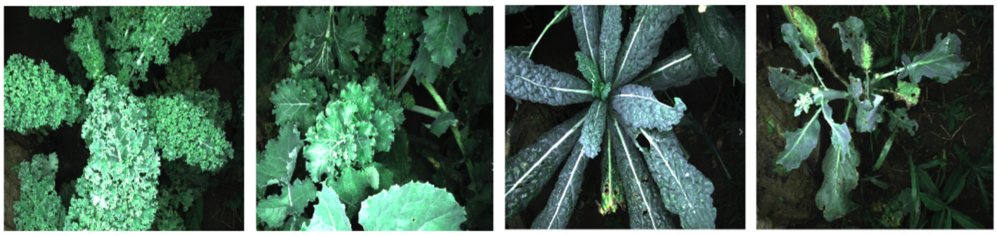


Figure 1: Pipeline

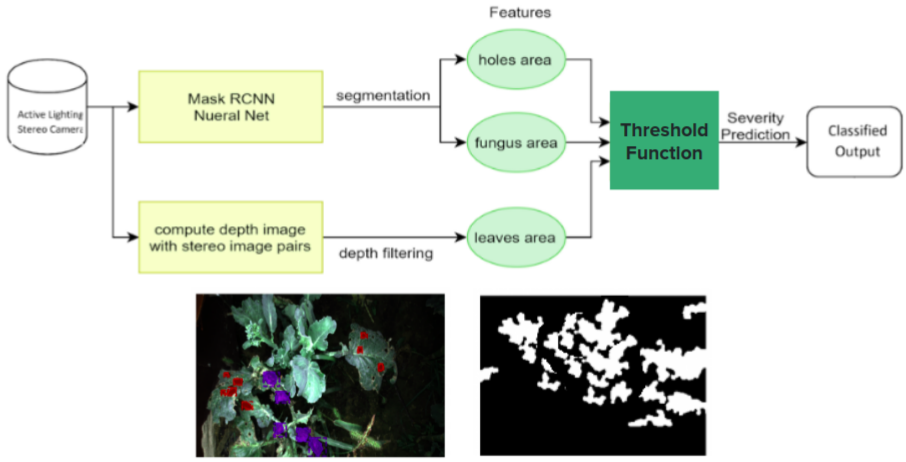


Figure 2: Plants to evaluate

1.1.3 Performance Metric

Given an image, the pipeline would determine if it detects (1) hole or (2) fungus. The task is defined as 2 binary classification tasks conducted at the same time. To evaluate the performance, a problematic symptom i.e. presence of holes or fungus is defined as positive. Furthermore, precision and recall are applied as metric.

Precision: $\frac{truepositive}{truepositive+falsepositive}$
Some intuition: when a symptom is reported for a plant e.g. presence of holes, what is the likelihood that the aforementioned symptom actually presents on the plant? A low precision indicates false alarm are common.

Recall: $\frac{truepositive}{truepositive+falsepositive}$
Some intuition: of all the plants that presents a problematic symptom e.g. holes, how many % of them are detected? A low recall rate indicates negligence of symptoms.

1.1.4 Parameters

Given an input image, some symptom pixels will be segmented out by the Mask-RCNN model. A threshold function is applied with various threshold values to evaluate the difference in performane. The same test is conducted for confidence level* = 0. 1 and 0.7.

The lower the confidence level, the more symptoms a Mask-RCNN model will report. But there will also be more false samples e.g. robot tire mis-classified as fungus presented in the image.

1.1.5 Choosing an Optimal Threshold

Since the collected data from the field is very imbalanced (with far more problematic plants than healthy plants), equal error rate is applied to determine the optimal threshold (Fig 3). To be more elaborate, equal error rate means a point where false rejection rate (defined as % of wrongly neglected sample of all positive samples) equals to false acceptance rate (defined as % of wrongly reported positive-samples of all negative samples).

Consider the following example:

given a test dataset of 20 positive samples and 9 negative samples.
at threshold α:
4 false positive, 4 false negative
precision: 80% recall 80%

at threshold β:
3 false positive, 7 false negative
precision: 81% recall 65%

Although threshold α has a better precision/recall performance, roughly half of its negative samples are wrongly labeled as positive, and 20% of its positive samples are labeled as negative. Whereas threshold β has a equal error rate of 33% in both positive and negative samples.

Since there's no prior knowledge of the population of negative and positive samples on the field, equal error rate would served as an unbiased metric to evaluate the performance.

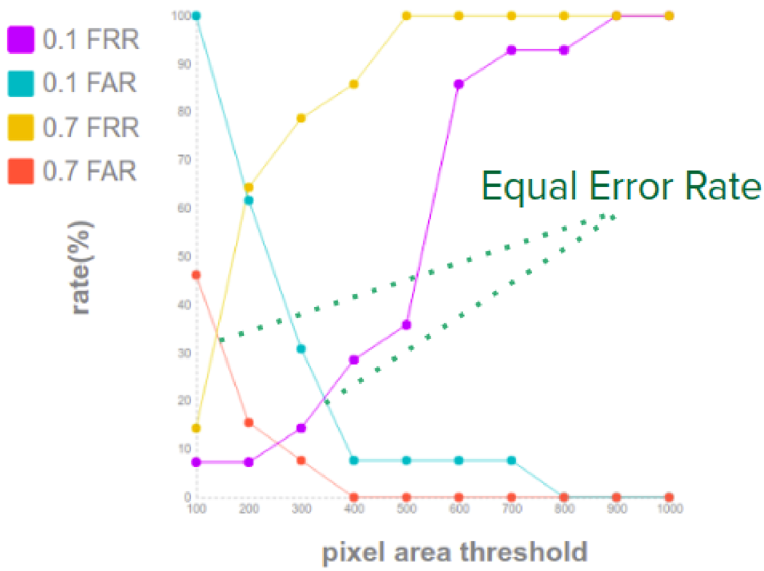


Figure 3: Equal Error Rate

1.1.6 Results

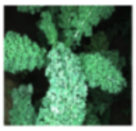
Roughly speaking, using pixel area directly instead of area ratio yields better solution (Fig 4 and 5). This is not surprising as estimation of leaf area introduces more noise and often times there's a 50% change observed in the leave estimation.

Note that for curly kale (the one with a lot of holes in nature) the precision and recall of hole for that specific plant is pretty bad. Therefore, this plant would be skipped for hole monitoring as it does not generate helpful information.

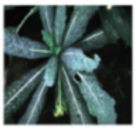
Performance Obtained with Pixel Area as Threshold

	conf	threshold	Precision (%)	Recall (%)
hole optimal	0.1	300	71	71
fungus optimal	0.7	300	91	84

	conf	threshold	Precision (%)	Recall (%)
hole optimal	0.7	200	82	82
fungus optimal	0.7	200	82	77



	conf	threshold	Precision (%)	Recall (%)
hole optimal	0.7	600	25	33
fungus optimal	0.1	600	81	81



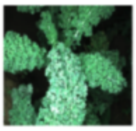
	conf	threshold	Precision (%)	Recall (%)
hole optimal	0.7	1000	85	61
fungus optimal	0.7	1200	75	75

Figure 4: Performance using Pixel Area

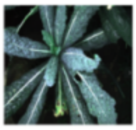
Performance Obtained with Area Ratio as Threshold

	conf	threshold	Precision (%)	Recall (%)
hole optimal	0.7	0.002	63	50
fungus optimal	0.1	0.008	86	86

	conf	threshold	Precision (%)	Recall (%)
hole optimal	0.7	0.006	80	70
fungus optimal	0.7	0.011	81	72



	conf	threshold	Precision (%)	Recall (%)
hole optimal	0.7	0.006	26	39
fungus optimal	0.7	0.006	87	77



	conf	threshold	Precision (%)	Recall (%)
hole optimal	0.1	0.006	85	66
fungus optimal	0.1	0.014	83	83

Figure 5: Performance using Pixel Area

1.2 Integrate Mask-RCNN with visualizer

The data flow from recorded ROS bag to graphic user interface is established for a row of real data collected from the field. The green dot indicates healthy while red problematic. (Fig 6)

2 Challenges

The main challenge came from modifying the architecture of Unet to get desirable results. Specifically in debugging how the network should be structured and formulating the cost function. These challenges can be tackled as John and Dung-Han Lee

Figure 6: GUI

both have experience with pytorch, and John can support in this task as similiar task have been conducted for him in the last semester.

2.1 Future works

debugging how the network should be structured, specifically making sure that the structure makes sense for a multi-class segmentation task. Also literature research and searching over stack overflow are needed to make sure a proper cost function is formulated for the task.

3 Teamwork

John has been working on debugging the localization with rows of plants with the orientation of the robot, specifically by projecting the RTK GPS measurement on to the heading direction of robot; John also works on revisiting LiDAR based navigation since the new farm layout is simpler for the task. John also supported Dung-Han Lee in Unet training.

Aaditya has been working closely with Hillel and Dung-Han Lee for multi-row visualization on a GUI. Specifically he has designed a new class to make the task more scalable. Meanwhile, Dung-Han Lee has supported him in integrating the pipeline and having the pipeline work on his computer.

Hillel has been working to add on features according to farmers feedback, as well as fabricating electronic wires for snesors and plant guards for filed test.

Aman has been working planner for a given farm layout, which would generate the planned trajectory for the robot to follow.