Progress Review 10

Individual Lab Report 9

Aaditya Saraiya

Oct 24, 2019

Team E:

Wholesome Robotics

Teammates:

Aaditya Saraiya

John Macdonald

Dung-Han Lee

Aman Agarwal

Hillel Hochsztein

Individual Progress

Capstone Project

Team E is creating an organic monitoring robot which has to autonomously navigate through crop rows. For this phase of the project, I had two main tasks which can listed as follows:

1) Version 2 of the results with the automatic image exposure script. The image exposure script is meant to check if the images which are collected are considered as acceptable images by the deep learning framework or not. The current requirement states that the number of acceptable images should be above 75% of the total images.

2) Work with Aman on setting up the robot_localization package and getting initial results with localization on recently collected ROS Bags.

Image exposure checking script

General idea of the problem

From the data collected during the previous semester, Dung-Han Lee had observed that a good percentage of images were either under-exposed or over-exposed. Over the time period, the reason of the problem was recognized to be variable heights of plants along the plant rows. In the previous PRs, initial work on auto-exposure based exposure control algorithms were discussed in order to solve this problem.

However, with the recent changes in the farm infrastructure the scope of the problem has reduced. In the newly planted crop rows, the height of the plants is pretty constant throughout the row. Hence, the team had decided to pause the work on auto-exposure algorithms and check the scale of the problem on the newly collected dataset.

The goal of this exposure checking script was to find out the percentage of images from any given ROS Bags which are under-exposed or over-exposed. The discussion on the requirements of the script as well as the tunable parameters has been done below.

Requirements of the exposure checking script

The requirements for the given exposure checking script can be summarized as follows:

1) The script should take a ROS Bag as input and output whether the images from the ROS Bag are usable or not.

2) The script needs to be automated as it will be integrated as a part of the full plant health monitoring pipeline.

3) The number of tunable parameters should be kept to the minimum and the exposure checking script should be able to generalize to different ROS Bags of different plant rows.

Keeping the core requirements of this script in mind, the following section will be a brief discussion on the general approach as well as tunable parameters in this script and how decisions on them were made currently.

Discussion on general approach and tunable parameters

This section briefly describes the approach taken to analyze the exposure of the image and make a decision on whether the image is over-exposed, under-exposed or normal.

Decision 1**:** What kind of model will be used to analyze the image statistics?

Currently, a histogram of image intensities is utilized in order to analyze the exposure of the image. The histogram of image intensities divides the image in 255 bins. Each bins consists of the number of image pixels in the image which have that particular value. If the general distribution of the images is under-exposed, than the histogram would be skewed towards the left side. Similarly, if the image is over-exposed, the histogram would be skewed on the right side.

Decision 2**:** Deciding the decision-making information

In this model, currently, the key factor which decides if an image is under-exposed or over-exposed is the number of pixels which are considered as 'under-exposed' or 'over-exposed'. An image is considered under-exposed or over-exposed if the number of pixels of those type are greater than 75% of the total pixels in the image. Currently, this number is decided by a threshold which has been decided from visual inspection by viewing the image histogram. However, the challenges with this approach will be discussed in the challenges section.

Decision 3**:** Tuning threshold and effect on results

Currently, a pixel is considered as under-exposed if its intensity value is below 40. An image is considered as over-exposed if its intensity value is 200. These thresholds have been arbitrarily chosen from visual analysis on the image data captured for a cabbage row at Rivendale farms.

Results and Comments on results

The following Figures (Figure 1 and Figure 2) showcase the results of the image exposure checking test. The script was run on 1162 recently collected images of a cabbage row. Figure 1 showcases the image exposure results with the under-exposed threshold set to below a value below 40 intensity. Figure 2 showcases the image exposure results with the under-exposed threshold set to below a value below 35 intensity. It can be seen that for this bag, a decrease in the threshold by an intensity value of 5 leads to the result changing from passing the requirements to failing the requirements. The challenges and issues with parameter tuning is discussed in the challenges section. Figure 3 shows the histogram of one image from this row. This image is representative of the general intensity statistics of the row. It can be seen that most of the images in this row are borderline under-exposed.

```
Under-exposed pixels= [211061.]
Normal pixels= [50761.]
Over- exposed pixels= [322.]
('Under-exposed pixels % ', '[0.8051338]')
('Over-exposed pixels % ', '[0.00122833]')
('Normal pixels % ', '[0.19363785]')
Done with image
1162
----------------------------

Total number of images 1162
----------------------------

Number of under-exposed images 200
----------------------------

Number of over-exposed images 0
----------------------------

Number of normal images 962
----------------------------

---------- FINAL RESULTS ------------------

Percentage of under-exposed images 17.2117039587
Percentage of over-exposed images 0.0
Percentage of normal images 82.7882960413
Yayy! We passed the test
```

Figure 1: Image exposure script results with an under-exposed threshold= 40

```
Under-exposed pixels= [221361.]
Normal pixels= [40461.]
Over- exposed pixels= [322.]
('Under-exposed pixels % ', '[0.8444252]')
('Over-exposed pixels % ', '[0.00122833]')
('Normal pixels % ', '[0.15434647]')
Done with image
1162
----------------------------

Total number of images 1162
----------------------------

Number of under-exposed images 449
----------------------------

Number of over-exposed images 0
----------------------------

Number of normal images 713
----------------------------

---------- FINAL RESULTS ------------------

Percentage of under-exposed images 38.6402753873
Percentage of over-exposed images 0.0
Percentage of normal images 61.3597246127
We failed the test
```

Figure 2: Image exposure script results with an under-exposed threshold= 35

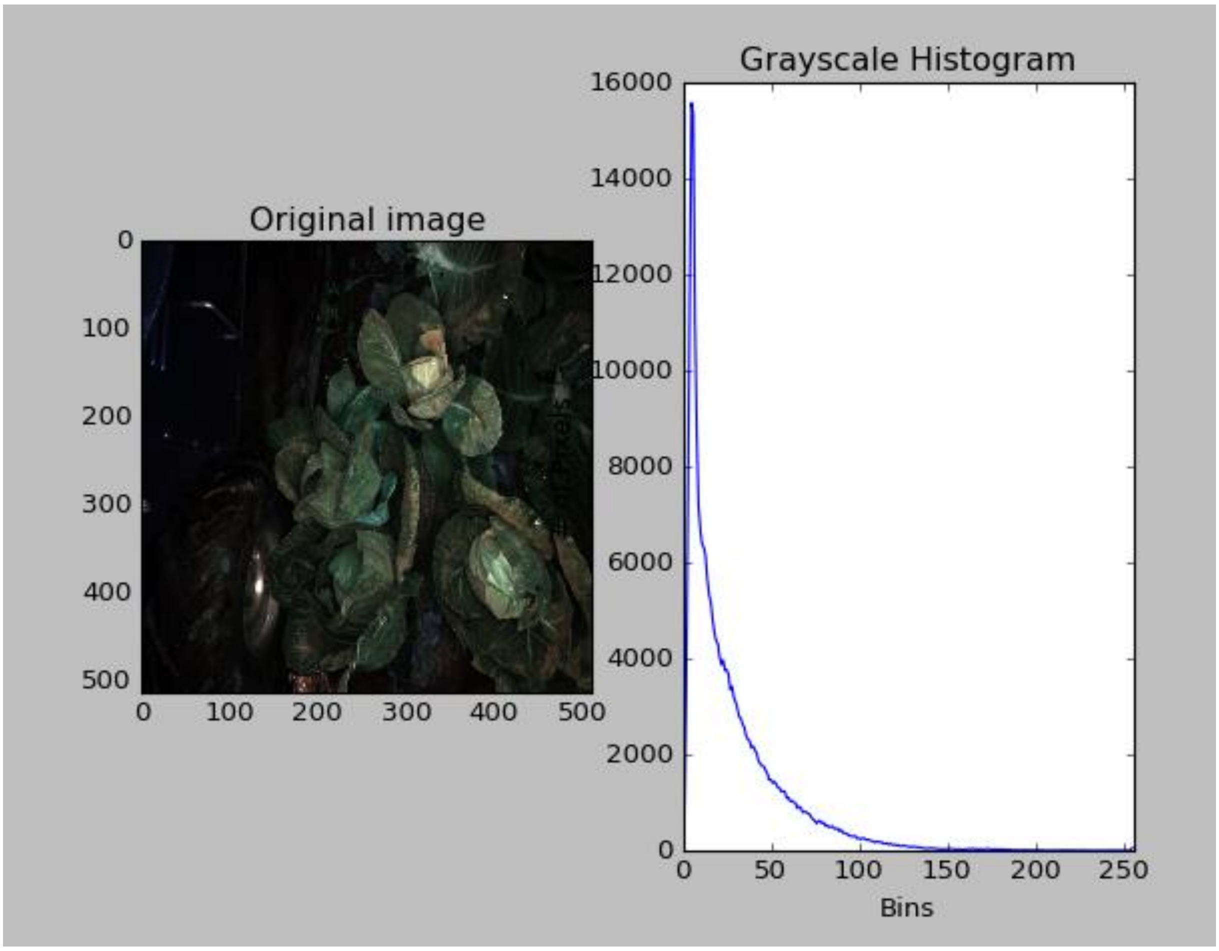

Figure 3: Histogram for a random image from the row

Conclusions

From the initial discussions, it can be seen that the change in threshold can cause considerable change in results. Hence, this is a matter of concern and if this whole process needs to be automated, setting this threshold would require more thoughts.

Secondly, it was recognized during the process of testing this script that there is a very strong human bias involved in setting thresholds. There may not be a direct co-relation between what humans find 'usable' and what the deep network considers usable. Even though the image statistics may conclude that the image is under-exposed, the deep network may not recognize that.

Further thought will be put on solving these minor issues in the following PRs when a full MVP of the monitoring pipeline will be demonstrated.

Working on Sensor Fusion for localization

This part of the project also involves improving the robot localization using sensor fusion with the robot_localization package. In coordination with Aman, the initial work involved reading through the package documentation as well as understanding at a higher level how the package is generalized for accepting 'n' inputs to receive state information. A thorough reading of the Generalized EKF [1] will be performed in the coming progress review.

Challenges

The key challenges faced during this progress review lie in the image exposure based checking script. During the start of the progress review, it was recognized that a histogram based method would be suitable for analyzing image statistics and making a decision on whether the image is over-exposed or under-exposed. While collecting data, we recognized that for the cabbage row, the linear actuator on the robot which had the camera mounted on it, was not able to go below a certain maximum level. Hence, during the field visit, we had concluded from visual inspection that the images seem to be okay. However, while testing the bag with the exposure script, it was observed that the images are borderline under-exposed. This leads to conclusion that some amount of image exposure feedback would be useful for the user for the field visit to conclude if the current camera settings are efficient or not. This feature will be potentially discussed and added before the next field visit.

Secondly, for the localization for sensor fusion, it was recognized that the documentation for the robot_localization package is very detailed and sometimes opaque. In some cases, we had to write extra ROS nodes as we had not collected the required rostopics during the field visit. Hence, the vast amount of parameter tuning required for the robot_localization package is a potential challenge and will need further analysis before the next progress reviews.

Teamwork

John Macdonald

John worked on improving the LIDAR based row segmentation model using the data collected from the last field visit. He worked on using an unsupervised learning based Gaussian Mixture model to look for improvements in the segmentation performance.

Aman Agarwal

Aman worked on doing initial analysis and trials on using the robot_localization package from ROS for performing sensor fusion between the IMU and visual odometry from the ZED camera.

Hillel Hochsztein

Hillel worked on adding support for multi-row clustering from the co-related image-location data. He also worked on integrating the already existing pipeline with the new multi-row features.

Dung-Han Lee

Dung-Han Lee worked on finding alternative methods for binary classification. In this case, he trained the UNet model on the new plant data and achieved higher precision and recall as compared to the previous Mask R-CNN based model.

Future Plans

Team

With the monitoring as well as navigation parts of the pipeline coming to maturity, this next PR will focus on solving the existing problems in the pipeline and showcase initial MVPs for both monitoring and navigation pipelines. The future team goals for the upcoming weeks can be summarized as follows:

1) Solving yaw estimation error by performing sensor fusion.
2) Iterations on the row detection and plant health detection modules by using alternative theorized models to improve the current performance of these

systems.

3) Complete the MVP of the monitoring pipeline and tie up loose ends in integration efforts.

4) Initial work on adding additional features in the plant health visualizer which are requested by the farmers on previous field visits.

Individual

The future individual tasks which are planned for the coming weeks can be summarized as follows:

1) In coordination with Aman, the main task for the next week includes working on the robot_localization package to perform sensor fusion for IMU, Visual odometry and RTK GPS data. This includes developing a strong theoretical as well as practical development of the underlying code in the robot_localization package.

2) Coordinating with Hillel and Dung-Han Lee to showcase version 1 of the plant health monitoring pipeline.

References

[1] Moore T., Stouch D. (2016) A Generalized Extended Kalman Filter Implementation for the Robot Operating System. In: Menegatti E., Michael N., Berns K., Yamaguchi H. (eds) Intelligent Autonomous Systems 13. Advances in Intelligent Systems and Computing, vol 302. Springer, Cham