Classification of plants in corn fields using machine learning techniques

by

Pruthvidhar Reddy Dhodda

B.E., Vasavi College of Engineering, 2016

A THESIS

submitted in partial fulfillment of the requirements for the degree

MASTER OF SCIENCE

Department of Computer Science
College of Engineering

KANSAS STATE UNIVERSITY
Manhattan, Kansas

2018

Approved by:

Major Professor
Dr. William H. Hsu

Copyright

Abstract

This thesis addresses the tasks of detecting vegetation and classifying plants into target crops and weeds using combinations of machine learning and pattern recognition algorithms and models. Solutions to these problems have many useful applications in precision agriculture, such as estimating the yield of a target crop or identifying weeds to help automate the selective application of weedicides and thereby reducing cost and pollution. The novel contribution of this work includes development and application of image processing and computer vision techniques to create training data with minimal human intervention, thus saving substantial human time and effort. All of the data used in this work was collected from corn fields and is in the RGB format.

As part of this thesis, I first discuss several steps that are part of a general methodology and data science pipeline for these tasks, such as: vegetation detection, feature engineering, crop row detection, training data generation, training, and testing. Next, I develop software components for segmentation and classification subtasks based on extant image processing and machine learning algorithms. I then present a comparison of different classifier models developed through this process using their Receiver Operating Characteristic (ROC) curves. The difference in models lies in the way they are trained - locally or globally. I also investigate the effect of the altitude at which data is collected on the performance of classifiers. *Scikit-learn*, a Python library for machine learning, is used to train decision trees and other classification learning models. Finally, I compare the precision, recall, and accuracy attained by segmenting (recognizing the boundary of) plants using the excess green index (ExG) with that of a learned Gaussian mixture model. I performed all image processing tasks using OpenCV, an open source computer vision library.

Table of Contents

List of Figures

List of Tables

Acknowledgements

First and foremost, I would like to thank my major professor, advisor, and mentor, Dr. William H. Hsu for his invaluable help, support and motivation throughout my graduate studies. I admire him for the time and effort he dedicates to academia. There is always something to learn from him. Joining his research group has had a very positive effect on me.

I thank Dr. Ignacio Ciampitti who gave me the opportunity to work on this project. I am thankful to him for being very kind and understanding with me. Without his support and motivation this work would not have been published in Remote Sensing journal.

I would also like to thank Dr. Mitchell L. Neilsen for dedicating his valuable time to serve on my committee. I am always amazed by the quality of research that happens under his supervision. I feel honored to have all of them on my committee.

I would like to thank Mr. Sebastian Varela for constantly pushing me to achieve better results and for his contribution in getting this work published. I also thank my mom, Uma R. Gurrala, dad, Manmadhar R. Dhodda, sister, Susmitha R. Dhodda, and my uncle, Maheshwar R. Gurrala for their immense love and support. Special thanks to my friends Shravani, Manoj, Poojitha, Pavan and Gnyani for being there during my ups and downs. Nithin, Sindhu, Sravani, Sneha, Sharmila, and Raja made my 2 years at K-State all memorable and fun. Finally, I would like to thank everyone who has been part of this wonderful journey.

Chapter 1 - Introduction

Corn (*Zea Mays L.*) is one of the most responsive crops to agronomic management practices (Lauer & Rankin, 2004). It has a limited capacity to compensate for missing plants within a row, and hence it is important to get early season stand count thereby helping farmers for efficient planning of operations such as re-planting decisions. Automatically determining not only plant count but distinguishing weeds from corn plants by classification is a crucial step which helps to minimize pollution by allowing site-specific application of weedicides, saving money which would otherwise be invested in weedicide application over the whole field. Visual inspection is a frequently used practice to get stand count which is labor intensive, time consuming, costly, and error-prone. There is a need for better approach which is fast and requires little or no human intervention. This work aims to solve the problem at hand using image processing and machine learning techniques. The workflow developed requires minimal human intervention in the training data creation which would otherwise require lot of labor in labelling the data.

The data consists of RGB images collected using Unmanned Aerial System (UAS) from corn fields in practical conditions from two different locations. They are captured in top-down view so that they are free from perspective projections. All other conditions like weed pressure, illumination, crop row density, etc., are uncontrolled. There is no restriction on the orientation of crop rows as the proposed methodology can deal with any orientation of crop rows with respect to x-axis.

A workflow is developed that generates training data by automatically labelling input data. It is possible with an assumption that states "all contours that lie in the row region are most probably corn plants and all other contours are most probably weeds". Feature engineering is done to extract features from input data. For this machine learning task, with its ground features, different decision tree classifiers are developed in two modes: local and global. In local mode, the model is trained and tested on data from the same location, while in global mode, the model is trained on data from both locations but tested on data from single location at a time.

These models are compared using receiver operating characteristic (ROC) curves. The effect of the altitude at which these images are captured on the performance of classification models is also presented using ROC curves.

Chapter 2 - Related Work

There have been a lot of improvements recently in ground sensors and computer vision which made plant counting easy in the field of proximal sensing. Using proximal sensing the scope for automation and mechanization improved drastically hence reducing the cost of plant counting. Srestha and Steward (2005) presented the use of size and shape of corn plants to estimate several attributes like plant density, row spacing by using video frame sequencing, segmentation and object classification. In a related context, Shi, Wang, Taylor, Raun, and Hardin (2013) proposed the laser line-scan technique to measure stalk locations during corn mid-growth stages. In all these methods proximal sensors are attached to ground vehicles to collect data like images or videos. However, the reachability of ground vehicles is limited to the terrain conditions and the robustness of vehicle being used. Ground vehicles also cause some damage to the crops in the fields and are limited to small areas. For effective weed management, there is a need for detailed knowledge on the spatial distribution of both crops and weeds. For these reasons, use of unmanned aerial systems is gaining importance. Proposed methodology uses UAS for collecting data (top down view imagery) from corn fields.

Burks, Shearer, and Payne (2000) proposed a new method to classify different weed species using color texture features for selective herbicide treatment. They were able to classify between five species of weeds and soil using hue and saturation statistics. In a similar context, Wu and Wen (2009) proposed the use of support vector machine as a classifier using texture features, principal component analysis was done to reduce the dimensionality. Although most of the related works proposed the use of machine learning to solve the problem in hand, only texture features are considered which might not capture the whole picture and don’t uniquely represent the corn crop or weeds. The proposed methodology in this work makes use of the fact that corn plants have unique geometric shape compared to the shape of possible weeds in the field. The features used to train the classifier are based on geometry rather than texture.

Another important step presented in this work that builds upon relevant prior work is row detection. There are several approaches to detect rows in a field. Varshney (2017) used segmentation to separate vegetation from the background using K-means clustering, Gaussian mixture models, support vector machine, Excess green index algorithm and fully connected convolutional neural networks. Then, they applied Hough transforms to find the orientation of crop rows. Green pixel accumulation is done, and a row template is fitted to identify row regions. The limitation of the workflow is that a fixed width template is fitted against possible crop rows whereas in the current workflow, an inter-row mask and row-mask are generated which function more like an adaptive template to match the width of individual crop rows. The current workflow has a step which automatically labels the data without much human intervention, saving a lot of time and effort which would otherwise be wasted in manually labeling the data.

Chapter 3 - Materials and Methods

3.1 Data Collection Sites

To make the methodology more useful for practical purposes all data collected and used for this work is collected in practical conditions and not from controlled environments like experimental sites. Different crop growth stages and crop residue were some of the field conditions present during data collection. Two fields were selected in the northeast region of Kansas, the US for data collection. One site (Site 1) was located in Atchison County, KS and the other site (Site 2) was located in Jefferson County, KS. Site 1 was 18 hectares in area and Site 2 was 64 hectares in area. Site 1 was managed using water from rainfall and Site 2 was under irrigation. Both the fields had an approximate plant density of 7.5 plants m^{-2}.

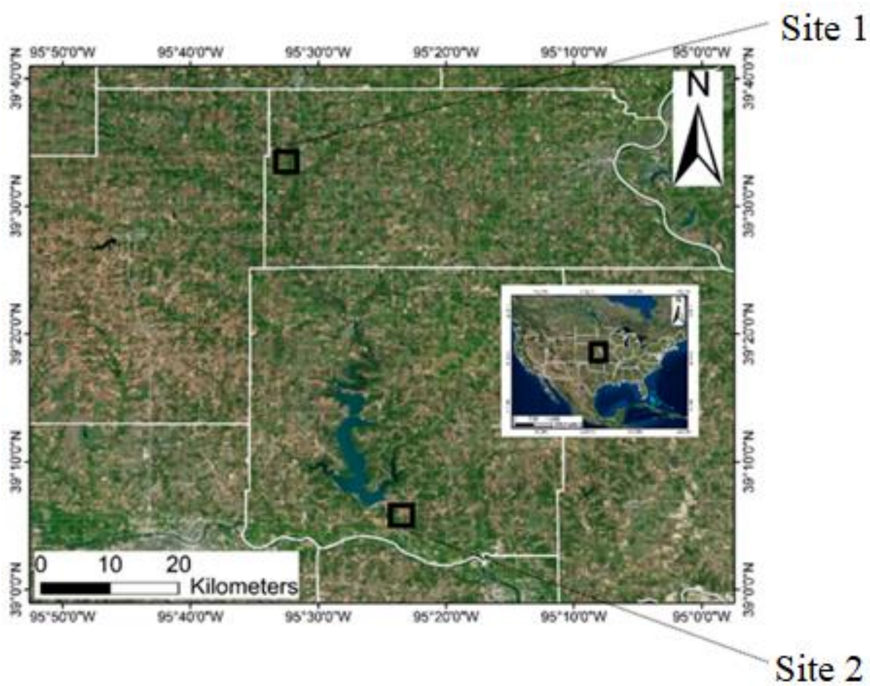

Figure 1. Adapted from (Varela, et al., 2018), Locations of Site 1 and Site 2

3.2 UAS, Sensor, and Data Collection

An octocopter, the S1000 by Dà-Jiāng Innovations (DJI) was used to collect the imagery from both sites. It has an A2 flight controller which is a multirotor stabilization controller that ensures the stability of UAS throughout the flight which in turn helps to capture images consistently. Out in the field, the conditions might be tough in terms of air currents which is undesirable for collecting data as the projection every time an image is captured changes. Thus, having the A2 flight controller is advantageous. It is also equipped with a Global Positioning System (GPS), enabling it to carry out automated flight missions using location co-ordinates. An autopilot software package called *PX4 Pixhawk Autopilot 2* is installed in the UAS complimentary to the Mission Planner Ground Station, an open-source software developed by Michael Osborne.

To account for diverse conditions, nine sample areas from each field are marked and data is collected from all these sample areas. Size of each of these areas is around 0.2 hectares. Each flight mission is programmed such that, UAS moves in four parallel lines capturing images for every 4 seconds, finally achieving 25 to 30 images per sample region. During the flight, overlapping and side-lapping was set to 20%. If overlapping is further increased the flight time increases thereby decreasing the efficiency of data collection process.

The camera used was SONY, Alpha ILCE A5100 RGB, mounted with a SONY SELP1650 PZ 16-50 mm lens which makes the sensor resolution 6000 x 4000 pixels. The exposure and aperture settings were adjusted before each flight to suit the conditions at the time of flight. The shutter speed was set to 1/3000 seconds, aperture to f5, focal length to 16 mm and ISO to 400. Wind speed was 2 to 3 meters per second when the data was collected from both sites. On the day of the flight, Site 1 has 2 visible leaves growth stage and Site 2 has 2-3 leaves growth stage. The altitude for the flight was set to 10 meters above ground level, thus achieving a spatial resolution of 2.4 mm.

3.3 Workflow

Figure 2 depicts the workflow that we implemented. Data (images) is collected in RGB format. Then each of the images is converted to a binary image using segmentation algorithms (ExG, GMM). After segmentation Hough line transformation is applied to find the orientation of crop rows which is then used to rotate images so that crop rows are nearly horizontal. Then, row and inter-row regions are detected using inter-row-mask and row-mask respectively which is used for scaling up the process of labeling data. Feature extraction, training, and testing are then performed whose results are discussed in Chapter 4. All the steps mentioned above are explained in detail in the following sections.

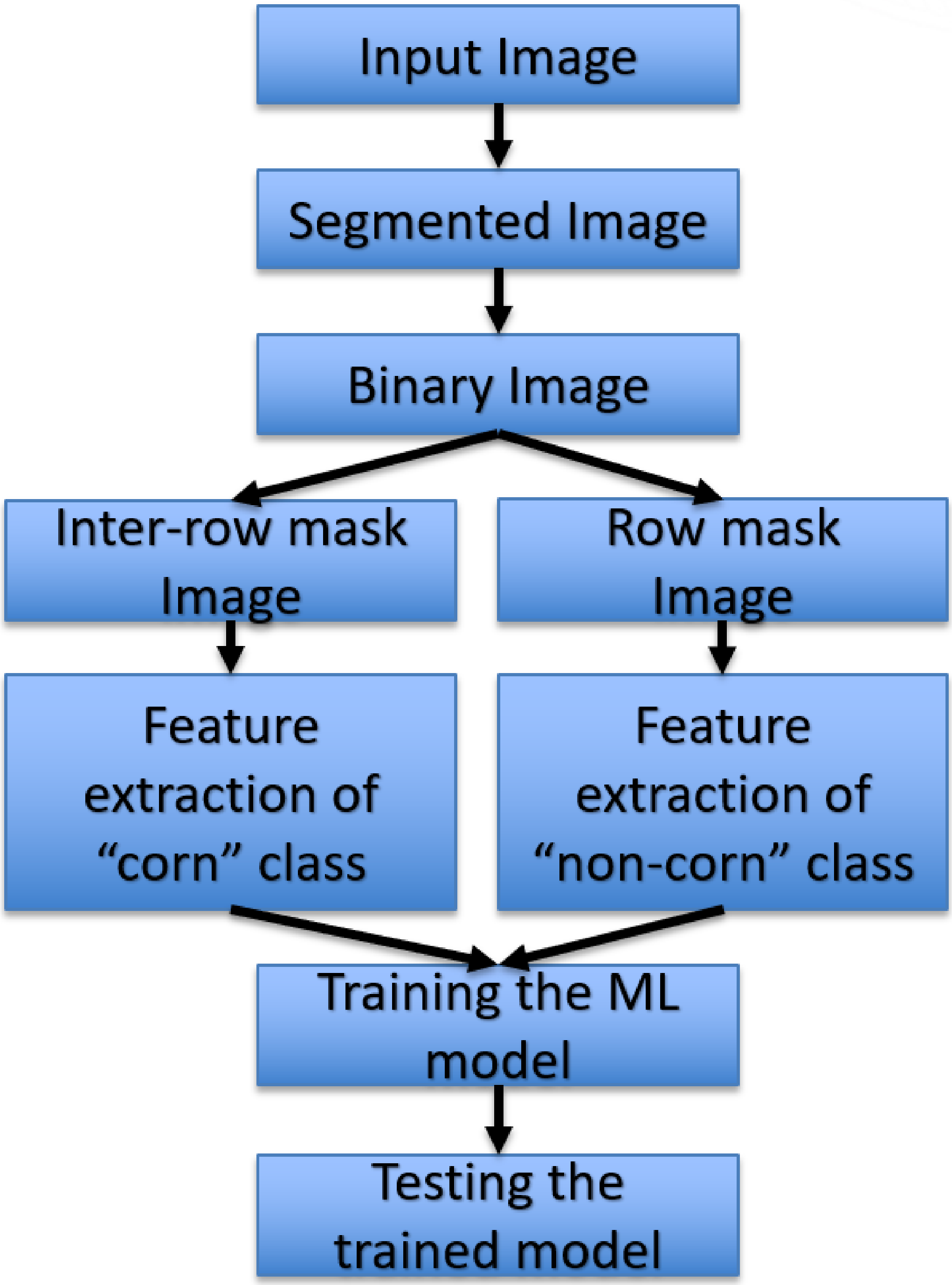

Figure 2. Workflow

3.4 Preprocessing Steps

Several preparatory transformations must be applied to the data before training a classifier with it. The steps that refine the raw data to be ready for training are discussed in this section.

3.4.1 Segmentation

Excess green index algorithm has been proven in the past to be a good algorithm for segmentation of green pixels and background (Woebbecke, et al., 1995). It differentiates green objects from the rest in an image. So, ExG has been used for segmentation of vegetation and background. Before that, to eliminate noise in the input images, a bilateral filter is applied which reduces unwanted noise while keeping edges sharp (Carlo & Roberto, 1998). Keeping edges intact is important as we want our segmentation to be as good as possible. Any changes to the edges might cause the geometric features to be not able to represent the actual shape of objects. Then the ExG algorithm is applied which computes ExG index for each pixel in the input image. ExG index helps to decide if a pixel corresponds to vegetation or background. It is computed as follows:

$$ExG = 2G' - R' - B'$$

where

$$G' = G/(R + G + B)$$

$$R' = R/(R + G + B)$$

$$B' = B/(R + G + B)$$

ExG represents the excess green index value of a pixel, R is the corresponding intensity value of that pixel in the red channel, G is the corresponding intensity value of that pixel in green channel and B is the corresponding intensity value of that pixel in the blue channel.

Then, a small modification (scaling) is done to improve the segmentation process. It is done as follows:

$$ExG' = \begin{cases} 255 & if\ ExG > 1 \\ ExG * 255 & if\ ExG < 1 \\ 0 & if\ ExG < 0 \end{cases}$$

ExG' is the modified Excess Green Index calculated using values obtained from Excess Green Index.

After getting the ExG' value of all pixels in the image, it needs to be thresholded in-order to have only two classes of pixels (foreground and background) in the image intensities: 255 for pixels that belong to vegetation and 0 for all other pixels. So, it results in a black and white image. The correct threshold value is determined by using Otsu's thresholding method which automatically decides the best threshold value (Otsu, 1979). It calculates the best threshold value to separate the two classes so that the intra-class variance is minimal. The weighted sum of variances of two classes which should be minimized is given by:

$$\sigma_w^2(t) = \omega_0(t)\sigma_0^2(t) + \omega_1(t)\sigma_1^2(t)$$

Where weights ω_0 and ω_1 are the probabilities of two classes of pixels separated by threshold t, σ_0^2 and σ_1^2 are variances of these two classes.

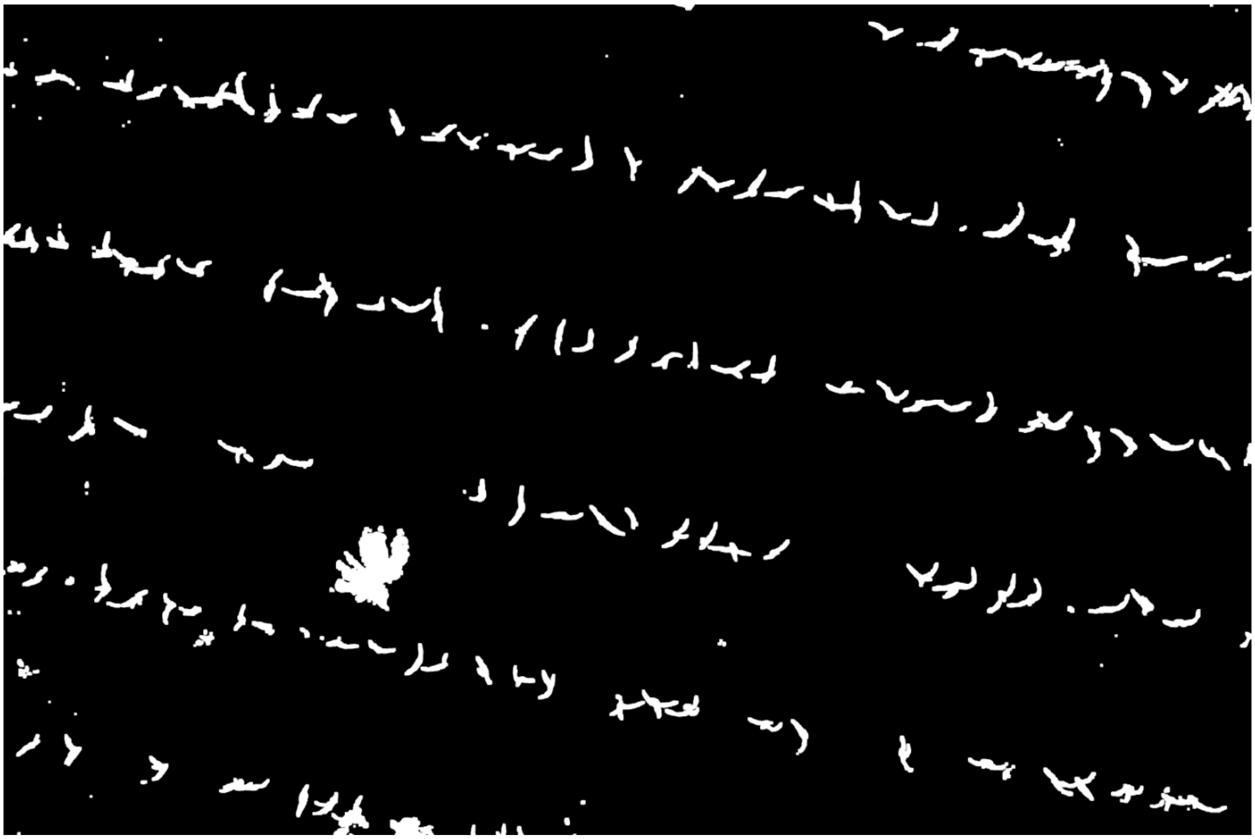

Figure 3. Image obtained after applying ExG segmentation and Otsu's thresholding

Besides ExG', Gaussian Mixture Models (GMM) is also used for segmentation process. Before fitting the data with GMM, to eliminate noise, a gaussian filter with kernel size (55,55) was applied. Then, data was fitted to GMM with two mixture components as our target was to differentiate between vegetation and background. The covariance type used was full. The results of using GMM for segmentation leaving the other workflow the same, are presented in Chapter 4.

3.4.2 Removing Big Contours

Now that we have a thresholded image of vegetation and background separated into white and black pixels respectively, the next step is to identify crop rows. Note that the pixels corresponding to weeds are also classified as vegetation pixels. From now on all the white pixels that are connected are collectively called contours. The current task is to remove contours that are too big, which might most probably correspond to weeds, so that they won't be counted as potential crop rows in the next step. If not removed, they result in a peak in greenness accumulation in next step which then will be considered as potential crop rows. Hence this step acts as a filter that removes very big patches of weeds. By experimentation, it is found that contours that are four times greater than average contour size in an image will most probably be a weed. So, all contours whose size is greater than four times the average contour size are removed by assigning intensity value same as background (0). These contours will be restored after Hough line transform is performed.

3.4.3 Crop Row Orientation

Now that we have a thresholded image having white contours that represent vegetation and rest all black pixels, the next step is to identify the orientation of crop rows. For this purpose, Hough Line Transform is used to first identify all possible lines that could exist in the image (United States of America Patent No. 3,069,654, 1962). There is a threshold value on the minimum length of the line in order for it to be considered. Through experimentation, it is decided that the optimal value should be around 150. After the algorithm is applied, set of lines and their orientation is returned.

Instead of averaging orientations of all the lines, a voting algorithm is developed which finds the best possible orientation. It is found that the returned orientation values might be only positive or only negative or both. In order to determine which is the right orientation, all the negative values are stored in a set and all the positive values are stored in another set. Now the set with a maximum number of elements and whose range is less than 20 degrees is selected. The average of all the values in this set is determined as the orientation of crop rows. If for an image, such value can't be determined, then that image is discarded. This step is very important in the workflow as the quality of training data is affected by the accuracy of the determined orientation. This is because we train the model based on features of contours present in crop rows as corn and on features of contours present out of crop rows as weeds. Thus, a small error in the assumed orientation leads to some corn plants being learned as weeds and vice versa.

3.4.4 Crop Row Detection

Now that we have the orientation of crop rows, the next step is to detect all rows present in the image. To achieve this, first, the image is rotated so that the orientation of crop rows is nearly horizontal. This is done by rotating the image by the exact amount determined in the previous step but in opposite direction. After having the crop rows in the horizontal direction, the intensities of all pixels in each row are summed up and a graph is plotted. Sum of intensities in each row are plotted on x-axis and y-coordinates of corresponding rows are plotted on the y-axis and the obtained graph is smoothened. The resultant graph looks like a wave in the vertical direction where crests represent the presence of higher number of vegetation (white) pixels and troughs represents lower number of vegetation pixels or no vegetation pixels at all. The background pixels which are black contribute nothing to the values in the graph as intensity value of black pixels is zero. Thus, the graph represents the distribution of vegetation class pixels. Some of the peaks in this graph truly represent crop rows, not all because weeds present in inter-row regions also result in peaks. To differentiate between crop rows and all other inter-row peaks a threshold value is to be determined. Before finding this threshold, all points with the sum of intensities less than 10000 are adjusted to value 0, because it is found by experimentation that now crop rows will have a value less than 10000. But, this value changes with the height at which the images are captured. To find the threshold value a simple approach is used, by iterating through all the peaks, if the current peak is less than one-third of the previous peak then the threshold value is set to current peak value as it is highly likely to be a result of inter-row vegetation. The leftover peaks after thresholding represent the peaks corresponding to crop rows. The width of each crop row was equal to the width of its crest at the thresholding region. Thus, the width of each crop row was determined dynamically unlike previous work (Varshney, 2017), where the width of each row was forced to be constant because of fitting a rectangular template.

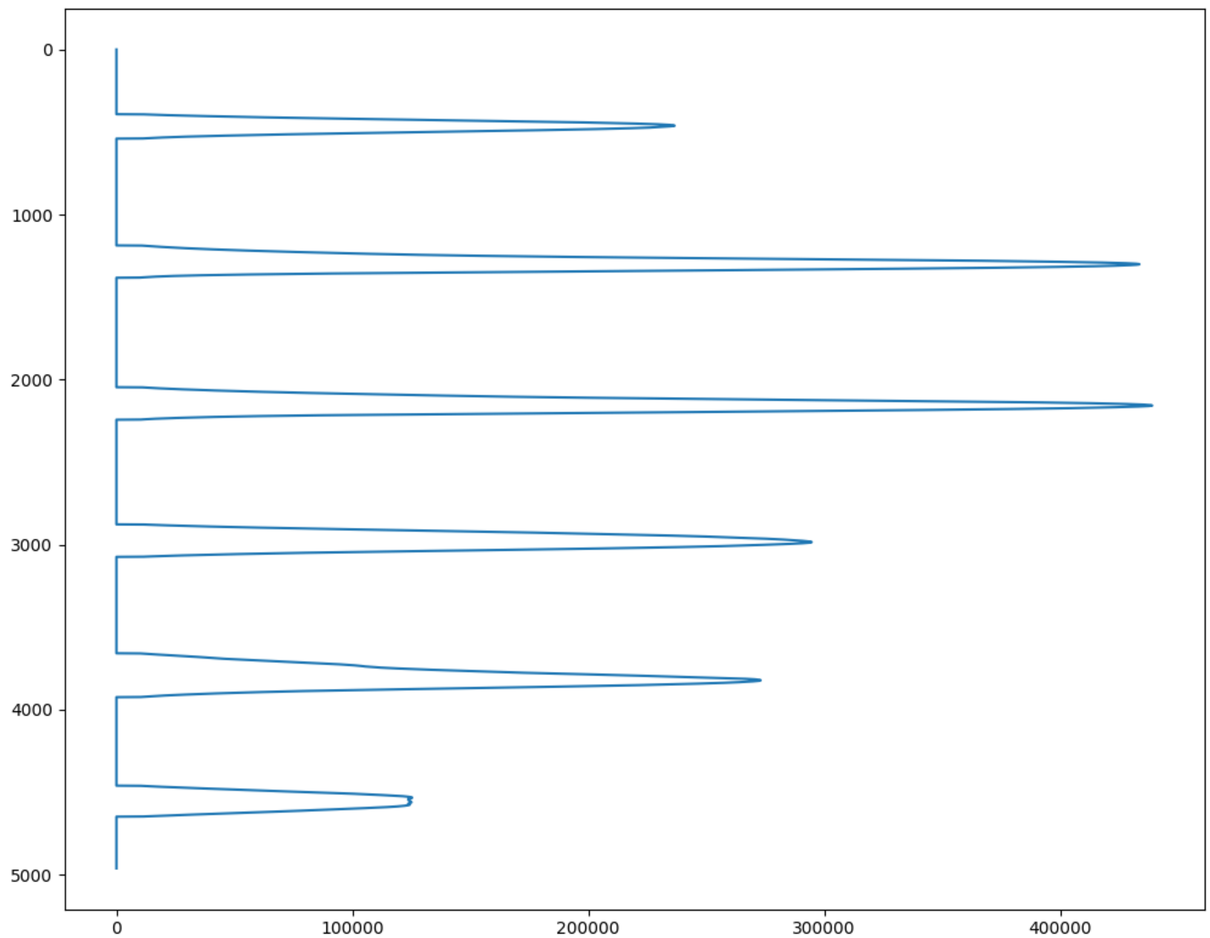

Figure 4. Row peaks with the sum of intensities along x-axis and height of the image along the y-axis

Using the outcome from the previous step, a mask (say inter-row-mask) was created matching the dimensions of the rotated image and has black rectangular strips in place of inter-row regions and white rectangular strips in place of crop row regions. Similarly, another mask (say row-mask) was generated by performing Bitwise NOT of inter-row-mask.

3.4.5 Training Data Generation

The novelty of this work is that there is no need for manual labeling of data. Instead, a Bitwise And operation is performed between inter-row-mask from the previous step and the rotated image, contours are then detected from the resultant image which are then labeled as “corn class”. Similarly, Bitwise And operation is performed between row-mask and rotated image, contours detected from the resultant image are labeled as “non-corn class”. With some exceptions, this generates a good labeled data set as there is a high chance of the presence of corn plants in row regions and a high chance of the presence of weeds or non-corn objects in inter-row regions.

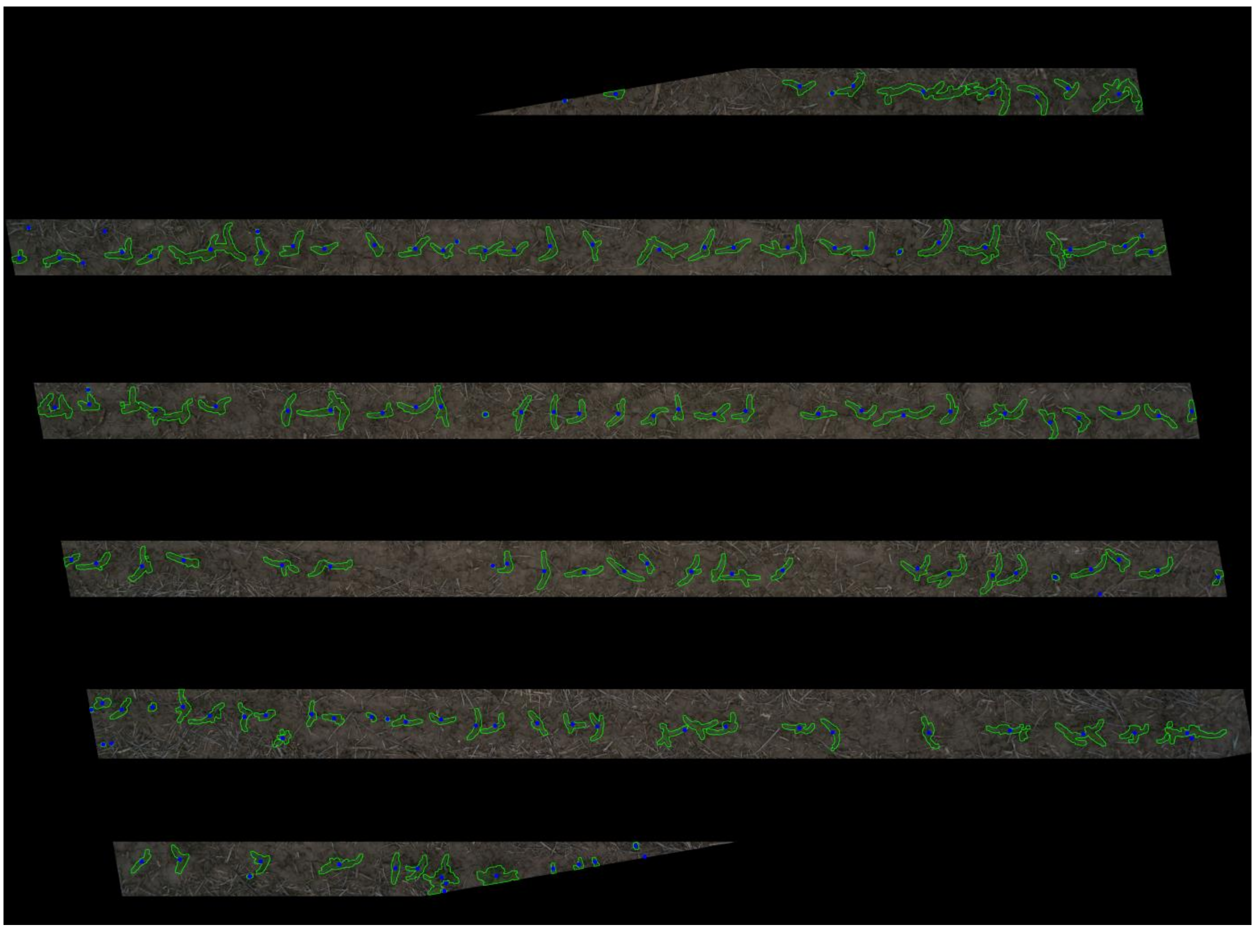

Figure 5. Result of performing Binary And operation on Inter-row mask and rotated input image

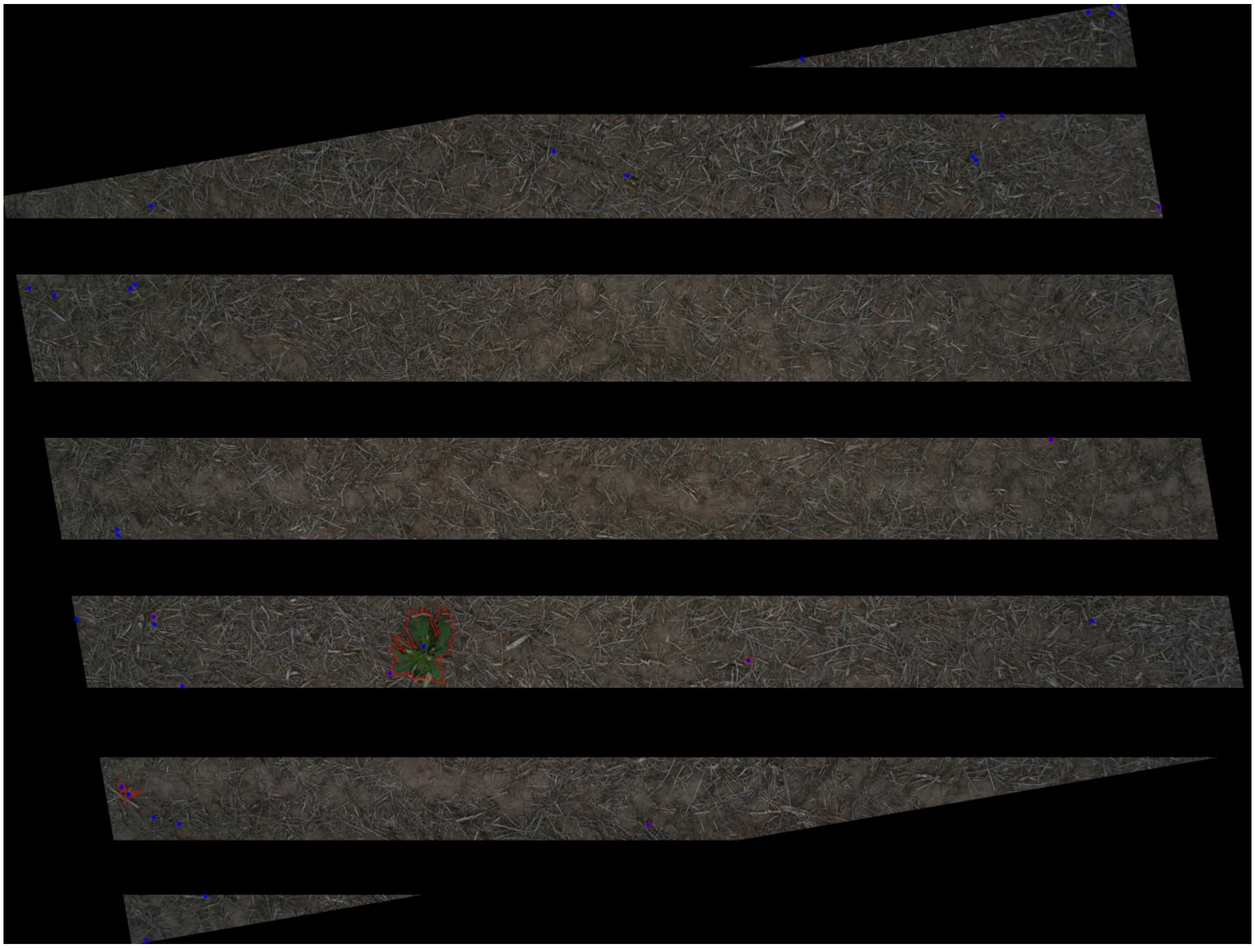

Figure 6. Result of performing row-mask and rotated input image

3.4.5.1 Feature Engineering

Since the contours were labeled, next step was to extract geometrical features from contours of both classes. Initially, nine geometrical features were considered but after applying a feature selection method they were filtered down to five features: aspect ratio, rectangularity, thinness, solidity and major axis to diameter ratio.

- Aspect ratio: It is the ratio of width to height of rectangle with minimum area enclosing the contour.
- Rectangularity: It is the ratio of the area of contour to the area of rectangle with minimum area enclosing the contour.

- Thinness: It is directly proportional to the ratio of the area of contour to the perimeter of contour.
- Solidity: It is the ratio of the area of contour to the area of convex hull enclosing the contour.
- Major axis to diameter ratio: It is directly proportional to the ratio of major axis length of contour to the square root of contour's area.

3.4.6 Training and Testing

A decision tree classifier was implemented, training with above mentioned geometric features and corresponding labels from training data generation step. Decision trees are supervised learning methods used for classification and regression (Breiman, et al., 1984). They come under non-parametric machine learning algorithms. They predict the target variable value by learning decision rules extracted from training data. There are different types of decision tree algorithms but the one used in this work was CART. CART uses information gain as a criterion while determining which attribute should be placed at which level of the tree. There are many advantages with using decision trees: capable of fitting a large number of functional forms which means it is flexible, cost of using is logarithmic in the number of training data points, performs well even when its assumptions are a bit violated as it has weak assumptions.

Decision trees have their own disadvantages. They are vulnerable to overfitting which means they fit well to the training data but do not generalize data well. To avoid this overfitting issue there are some parameters that can be tuned to make the tree work well with new unseen data. One of such parameters is the maximum depth of the tree. By setting it to a value we restrict the tree from growing big (complex) in the process of overfitting to training data. A stratified K-Folds cross-validation was performed to determine the best hyperparameters for the classifier. It was determined that a maximum depth of 10 levels would work well. The other hyperparameter is minimum sample leaves. Very low value of minimum sample leaves will result in overfitting and a high value would prevent decision tree from learning decision rules. It was determined that setting minimum sample leaves parameter to 20 would work well. It was observed that the data is imbalanced as the data is collected in practical conditions where number of corn plants outnumber the number of weeds. To solve this issue of class imbalance, I have decided to adjust the weights of each class inversely proportional to corresponding class frequencies in the input data. This can be easily done in *scikit-learn* by setting the value of parameter 'class-weight' to 'balanced' (Buitinck, et al., 2013).

	Site 1		Site 2	
Dataset	Training	Testing	Training	Testing
Images	94	75	87	75
Contours	17,608	15,378	16,855	15,246

Table 1. Data sets used for training and testing

To evaluate the scalability of the classifier, training and testing was done in two modes: (a) local training and local testing (LTLT) (b) joint training and local testing (JTLT). In LTLT, the classifier was trained on training data from a location and tested on testing data from the same location. In JTLT, training was done on training data from both locations (Site 1 and Site 2) and tested on test data from both sites separately. The ground Truth for the test data set is generated by manually labeling the contours present in the output of the classifier. Results from this step are discussed in next chapter.

Chapter 4 - Results

This chapter presents results in three sections. The first section deals with the comparison of two decision tree models, one developed using ExG' in segmentation step and the other one developed using GMM in the segmentation step. The second section deals with comthe parison of decision tree models developed using local training and local testing (LTLT) and joint training and local testing (JTLT) training and testing modes. The third section deals with the effect of altitude at which images are captured on the performance of classifiers developed using this workflow.

All model comparisons are done using Receiver Operating Characteristic (ROC) curves. Comparison of ExG' and GMM steps is also shown in tabular form that has number of contours detected in ExG', GMM and ground truth.

4.1 ExG' vs GMM

Table 2 consists of ground truth of number of objects, number of objects detected (segmented) using ExG' and number of objects detected (segmented) using GMM.

	Ground Truth	ExG	GMM
Number of objects	417	391	319

Table 2. Number of plants detected using ExG' and GMM.

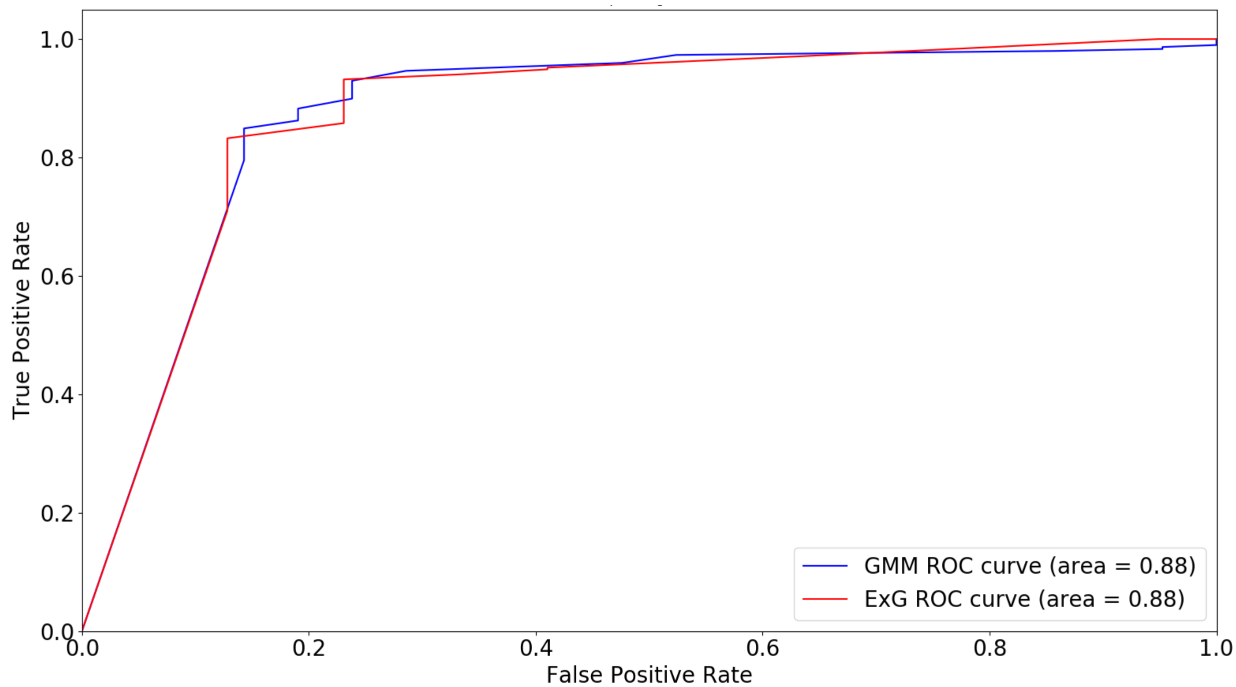

Figure 7. ROC curves of classifiers with GMM and ExG' used in segmentation step

	Precision	Recall	F1-score	Accuracy
ExG	0.93	0.92	0.92	0.91
GMM	0.94	0.89	0.91	0.89

Table 3. Classification report of classifiers with ExG' and GMM used in segmentation step

4.2 LTLT and JTLT

In this section, comparison of LTLT on site 1, LTLT on site 2, JTLT on site 1 and JTLT on site 2 is presented in the form of ROC curves. The precision, recall, and F1-scores are also presented for each of the models.

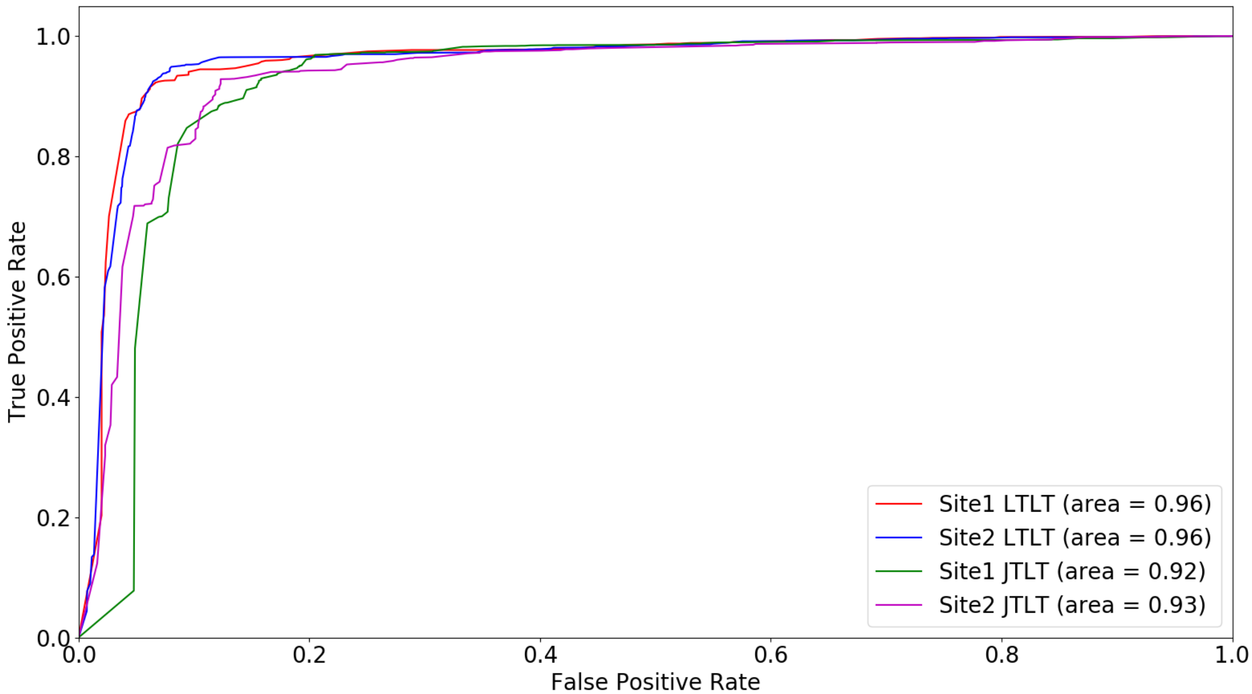

Figure 8. Adapted from (Varela, et al., 2018), ROC curves for classifiers trained on Site 1 and Site 2 data using LTLT and JTLT modes

	Precision	Recall	F1 Score	Area Under Curve
LTLT Site 1	0.93	0.93	0.93	0.95
LTLT Site 2	0.94	0.94	0.94	0.95
JTJT Site 1	0.93	0.93	0.93	0.92
JTJT Site 2	0.92	0.92	0.92	0.93

Table 4. Classification report of classifiers trained on Site 1 and Site 2 using LTLT and JTLT modes

4.3 Effect of Downsampling

Downsampling was done to investigate its effect on segmentation step and on the performance of the classifier. It helped us understand the tolerance of our workflow with degrading resolution (in a sense it is equivalent to increasing the altitude of flight while capturing images). The initial resolution of 2.4 mm in site 1 was changed to 4.8, 9.6 and 19.2 simulating 20, 40 and 80 meters flight altitudes respectively. As a result of this, the amount of loss in percentage of detected objects was found to be 6%, 12%, and 42%.

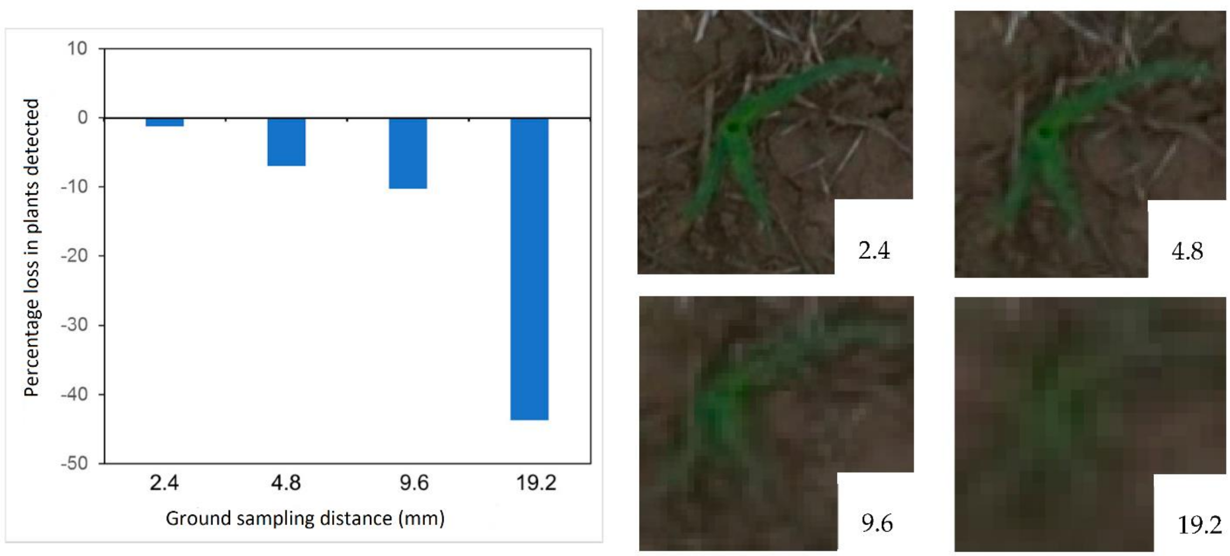

Figure 9. Adapted from (Varela, et al., 2018), Effect of downsampling on segmentation step

With decreasing resolution, the boundaries of objects deviated from their original shape affecting the geometrical features. So, this had its effect on the performance of classifier which is shown below.

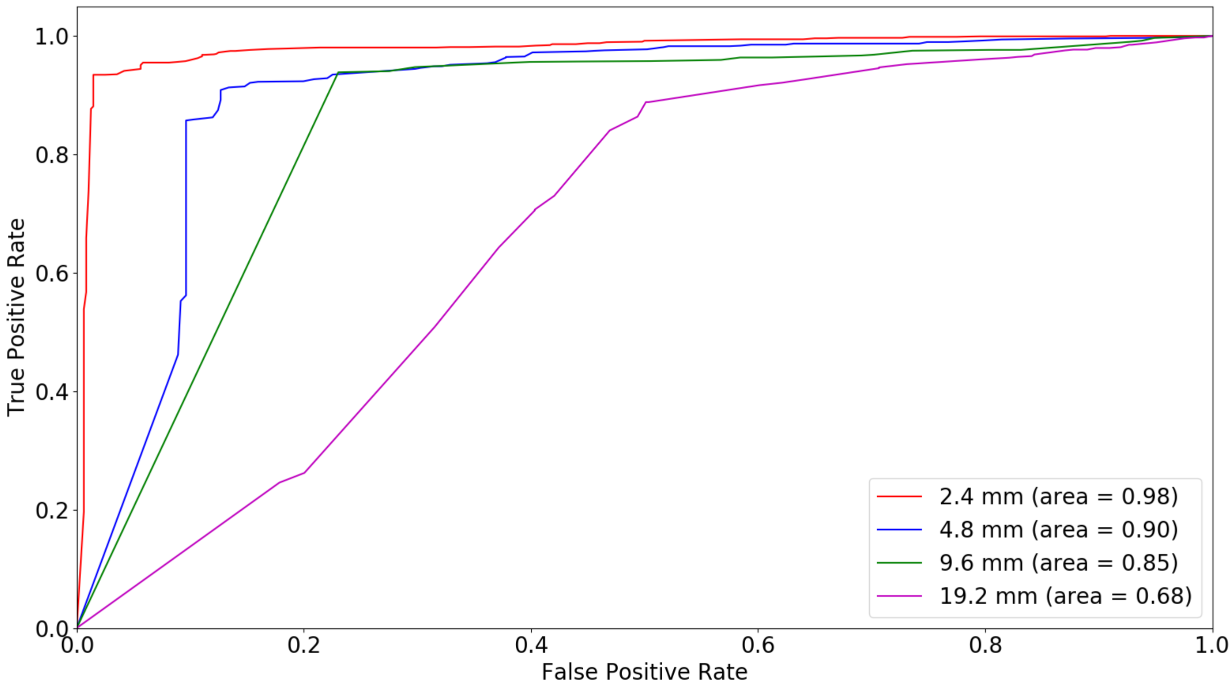

Figure 10. Adapted from (Varela, et al., 2018), ROC curves of classifiers trained using downsampled images up-to four levels

Chapter 5 - Conclusions

In this work, a workflow was implemented to segment plants in a corn field and classify them into corn and non-corn (weed) classes. Compared two techniques for segmentation step: Excess Green Index and Gaussian Mixture Models. Decision trees were used for the classification task and geometric descriptors were used as features for training the model. The novelty of this work is that, no or less human intervention is needed for the training process as the manual labeling step is replaced with automatic labeling step which scales up the training process.

The LTLT training mode had a slightly better area under curve value compared to JTLT training mode, a difference of 0.02 to 0.03 is observed. The difference is that, in case of JTLT false positives increased thereby decreasing the recall of "non-corn" class by 0.03. Same effect is observed in case of f1-score values of "non-corn" class but no such effect was observed in case of "corn" class. We can infer that LTLT performs better than JTLT. Since no or less human effort is needed in case of the training process, it is recommended to use LTLT training mode every time we need to use this workflow on the new data set.

We also observed that, better the quality of images, better is the quality of segmentation and classification task. It is understood that performance of workflow is improved at the cost of efficiency of flights (for data collection). Finally, Excess Green Index (modified) outperformed Gaussian Mixture Models in segmentation step. It is better to use ExG' as presented in this work for the segmentation task.

Chapter 6 - Future Work

The current workflow has some limitations: (a) It cannot deal with situations where corn plants overlap. (b) It cannot deal with situations where crop rows are not in a (nearly) straight line. (c) It is ineffective in presence of high weed pressure such that sum of intensities along the weed patch is greater than one-third of the sum of intensities of the previous crop row. Some of these limitations could be solved by using Region-Based Convolutional Neural Networks (R-CNN). So, building an R-CNN and training it from scratch with two classes (corn and non-corn) could be a good option to work in future. Before using deep learning techniques like R-CNN, it is a good idea to explore texture features along with geometric features so that another dimension in terms of features would be included.

The strength of this workflow lies in automatic labeling of training data thereby scaling up the training process, saving a lot of time and effort. This workflow can be improved by using segmentation algorithms such as the Watershed transform to solve the issue of overlapping plants. It can be extended to other crops by finding the correct set of features that can help distinguish target crop from weeds.

Chapter 7 - References

Breiman, L., Friedman, J., Olshen, R., & Stone, C. (1984). *Classification and Regression Trees.* Belmont, CA: Taylor & Francis.

Breiman, L., Friedman, J., Olshen, R., & Stone, C. (1984). *Classification and Regression Trees.* Belmont: Wadsworth.

Buitinck, L., Gilles, L., Mathieu, B., Fabian, P., Andreas, M., Olivier, G., & Varoq, G. (2013). {API} design for machine learning software: experiences from the scikit-learn. In *ECML PKDD Workshop: Languages for Data Mining and Machine Learning* (pp. 108-122).

Burks, T., Shearer, S. A., & Payne, F. A. (2000). Classification of Weed Species Using Color Texture Features AND Discriminant Analysis. *Transactions of the American Society of Agricultural Engineers*.

Carlo, T., & Roberto, M. (1998). Bilateral filtering for gray and color images. *Proceedings of the IEEE International Conference on Computer Vision*, 839-846.

Hough, P. (1962). *United States of America Patent No. 3,069,654.*

M. Woebbecke, D., Meyer, G., Von Bargen, K., & Mortensen, D. (1995). Color Indices for Weed Identification Under Various Soil, Residue, and Lighting Conditions. *Transactions of the ASAE*.

Otsu, N. (1979). A threshold selection method from gray-level histogram. *IEEE Trans. Syst. Man Cybern*, 62-66.

Shi, Y., Wang, N., Taylor, R. K., Raun, W. R., & Hardin, J. A. (2013). Automatic corn plant location and spacing measurement using laser line-scan technique. *Precision Agriculture*.

Srestha, D. S., & Steward, B. (2005). Shape and size analysis of corn plant canopies for plant population and spacing sensing. *American Society of Agricultural and Biological Engineers*.

Varela, S., Dhodda, P., Hsu, W., Prasad, P., Assefa, Y., Peralta, N., & Ciampitti, I. (2018). Early-Season Stand Count Determination in Corn via Integration of Imagery from Unmanned Aerial Systems (UAS) and Supervised Learning Techniques. *Remote Sensing*.

Varshney, V. (2017). *Supervised and unsupervised learning for plant and crop row detection in precision agriculture.* master's thesis, Dept. Computer Science and Eng., Kansas State University.

Wu, L., & Wen, Y. (2009). Weed/corn seedling recognition by support vector machine using texture features. *African Journal of Agricultural Research*.