Plant Phenotyping and Detection of Clouds and Shadows

Kevin Toney, Jackson Stansell, and Hessan Sedaghat

1 Introduction

According to the United Nations, the current world population is expected to reach 8.6 billion by 2030 and 9.8 billion in 2050 [1]. Although recent reports by the Food and Agricultural Organization showed the food production industry is growing by 1.2 percent every year, it is nowhere close to meeting the food demand that is expected to increase anywhere between 59% to 98% by 2050 [2]. To meet this need, agricultural producers are embracing innovative technologies to replace the traditional method of maintaining their crops and increasing food productivity. Traditional methods for acquiring crop traits, such as plant height, leaf color, chlorophyll content, biomass, and yield includes manual sampling, which is time-consuming and labor-intensive. As a result, Unmanned Aerial Vehicles (UAVs), equipped with different sensors, have become an important phenotyping tool in recent years. The aerial images obtained from UAVs are regularly used by crop researchers and agricultural producers to not only monitor crops during the growing season but also to make prompt and reliable judgments.

One of the many important decisions that are made after agricultural producers use UAVs is the amount of nitrogen fertilizers required to keep the crops healthy. One way to determine the nitrogen prescription uses the normalized difference red edge index (NDRE) from aerial images. NDRE utilizes the red edge band to detect the changes in chlorophyll content, thus determining the health of the crop. After measuring vegetation indices, such as NDRE, the nitrogen input is determined using the process described by figure 1.

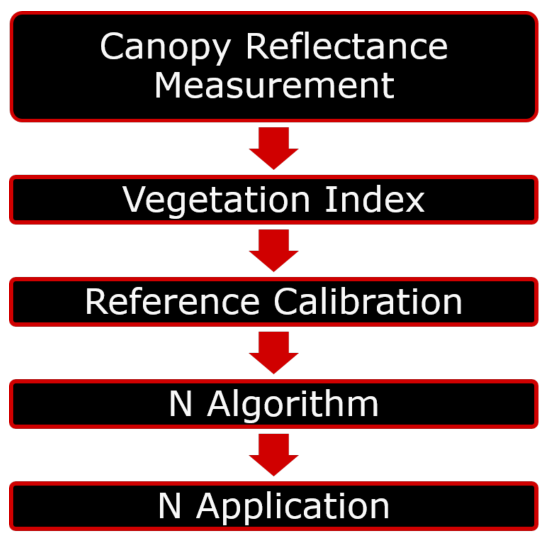

Fig. 1. Nitrogen prescription process. The crops' canopy reflectance is measured by a vegetation index. Then, the vegetation index is calibrated using a reference vegetation index. Next, the required amount of nitrogen is determined and applied to the field.

The indices calculated from aerial images are calibrated against a reference vegetation index to obtain the sufficiency index (SI). This index is derived using equation 1. The calculated SI is then applied to a nitrogen prescription algorithm, which is depicted by equation 2. This algorithm measures the amount of nitrogen that is required for the field. Hence, an inaccurate measurement of the NDRE will result in an inaccurate prescription of nitrogen. The implications of bad nitrogen prescriptions is one of the many reasons the accuracy of the NDRE data from aerial images is of utmost significance.

$$SI = \frac{NDRE}{NDRE_{ref}} \quad (1) \qquad N_{App} = N_{Opt}\sqrt{\frac{1 - SI}{\Delta SI}} \quad (2)$$

Hence, an inaccurate measurement of the NDRE will result in an inaccurate prescription of nitrogen. The implications of bad nitrogen prescriptions is one of the many reasons the accuracy of the NDRE data from aerial images is of utmost significance.

1.1 Problem Statement

Unfortunately, the accuracy of aerial images can be affected by various factors, one of which is the presence of clouds [3]. Clouds and their accompanying shadows are inevitable contaminants for aerial imagery. According to the estimation made by the International Satellite Cloud Climatology Project-Flux Data (ISCCP-FD), the global annual mean cloud cover is approximately 66% [4]. Clouds impede UAV's and satellites from obtaining clear views of the land. Moreover, the shadows cast by clouds eliminates crucial spectral information for the NDRE. As NDRE is influenced by clouds and shadows, ignorant farmers may give more nitrogen than they would on a regular day, resulting in wasted money and harmful effects on the environment. On the other hand, they may prescribe too small amounts of nitrogen, which hurt their crops in the long run.

The following paper explored ways to detect clouds and shadows in aerial images. Furthermore, this paper studied effects on the NDRE measurements. To accomplish these tasks the team used Full Vegetation Coverage (FVC) to estimate when the crop reaches full canopy. The team then developed an algorithm to detect clouds and trace them in aerial images. Lastly the effects of clouds and shadows were evaluated.

1.2 Potential Impact

The end goal of this study is to make input decision-makers aware of the significance of clouds and shadows in aerial images. This knowledge will enable agricultural producers to prescribe nitrogen to their crops in a more efficient and accurate manner. As a result, farmers will save money and produce healthier crops. Furthermore, this project may generate a discussion of how computer vision may improve the accuracy of nitrogen prescription recommendations delivered to decision-makers who work with corn, maize, soybeans and sorghum. Cloud and shadow detection could also be useful for solar energy production, and environmental monitoring.

For your information, the following project was done with accordance to the final project requirement for CSCE 473/873: Computer Vision at the University of Nebraska-Lincoln. The project was accomplished during the Fall 2019 semester.

2 Objectives

To accomplish the aforementioned project, three objectives were defined for this project:

1) Detect the full canopy stage: Approximate the date at which the crop reaches full canopy.
2) Detecting clouds and tracing shadows: Determine the presence of clouds and shadows over the fields in the images and trace their outlines.

3) Comparing data: Determine the factor that clouds and shadows influence vegetation indices and the sufficiency index.

3 DATASET

This project's images are part of a single dataset collected for an on-farm research project, which investigated site-specific nitrogen management via irrigation systems. The images are multispectral images. They were obtained from two different platforms - drones and planes. The drone images were collected from the aforementioned researchers. TerrAvion, a third party aerial imaging company, was the paid service provider for the plane-based aerial imagery. Figure 2 below shows the wavebands used in each dataset. The quantitative properties of the datasets are presented in Table 1.

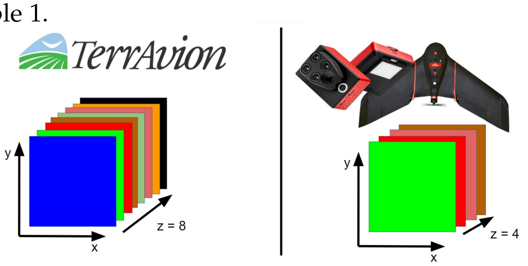

Fig. 2. The left image is a visual representation of the TerrAvion Images with 8 wavebands. On the right is a visual of drone images with 4 wavebands.

Table 1. These estimated values about the dataset are based on a completely square image.

Dataset Quantitative Properties							
Platform	Images	Bands	Dimension (pixels)[a]	Total Pixels[a]	Resolution (cm/pixel)[a]	Depth (bits)	Storage (MB)[a]
Drone	74	4	6500	4.2E7	12-15	32	150
Plane	71	8	5000	2.5E7	19	16	350

Each data set is discussed further below.

3.1 Drone Images

Drone images were captured with a Sensefly eBee SQ mounted Parrot Sequoia multispectral camera programmed to record images in the red, green, red-edge, and near-infrared bands. The camera had an individual image resolution of 12 cm/pixel. These images were taken from an altitude of approximately 120 meters. Individual images were "stitched" into a single mosaic that depicted the entirety of the field area using Pix4D, a privately owned and licensed software package. Additionally, Pix4D generated GeoTiff files that were used in this project. The GeoTiff images have variable x and y dimensional sizes, areal extents, and data sizes. Generally, the images covered greater than 80 acres in area. If the typical field were completely square, x and y dimensional sizes would be approximately 6500 pixels for a total of over 42 million pixels per image. Stitched images for each individual band, or derived index, have a resolution of between 12 and 15 cm/pixel at a 32-bit pixel depth. The stitched images took up roughly 150 MB of uncompressed memory. In the dataset, there were 74 drone images.

3.2 Plane Images

Plane images were captured with a small low-altitude airplane mounted multispectral camera, which captured images in 8 bands including red, green, blue, and near-infrared. Captured images encompassed the entirety of the interest area, so they didn't need to be "stitched" together like the drone images. Typically, the x and y dimensions of these images were approximately 5000 pixels, resulting in a total pixel count of 25 million. Image resolution is about 19 cm/pixel at a 16-bit pixel depth, which occupies roughly 350 MB of storage space. Overall, there are 71 of these 8-band images in the dataset. One of the advantages of the TerrAvion imagery was that they presented cloud identification labels on clouded images which could help evaluate the algorithm's performance.

4 Determining the Full Canopy Stage

4.1 Overview

The first objective was to determine when the corn fields reached full canopy. In other words, when did the corn crops cover the rows of the field. Upon investigation of current methods, the estimation of the full canopy can be done using two methods. These methods are the Leaf-Area index(LAI) and the Fractional Vegetation Coverage (FVC).

4.2 Related Work

First off, Leaf-Area Index (LAI) is a measure for the total area of leaves per unit ground area. LAI is directly related to the amount of light that can be intercepted by plants. This is a popular index for determining full canopy stage. There are three ways to measure LAI. First is using hand measurements. The problem with this method is that it is invasive and can damage the crops. The second method is performing gap fraction analysis, using digital images that has been captured from the ground looking up through the canopy [5]. The last method involves the estimation of LAI from aerial imagery [6] [7]. However, since we only had access to aerial images, not actual LAI measurements, we were not able to implement supervised learning methods to estimate the LAI. Regarding Fractional Vegetation Coverage (FVC), it is the ratio of vertically projected area of vegetation to the total surface extent. This ratio is generally expressed in relation to a unit area. This method is relatively much simpler to estimate compared to LAI [7]. Full Vegetation Coverage can be estimated using equation 3 where s represents the typical NDVI for bare soil and v is the typical NDVI for dense vegetation.

$$FVC = \frac{NDVI - NDVI_s}{NDVI_v - NDVI_s} \tag{3}$$

This estimation was called a linear mixture model. It can be obtained by taking the ratio of the difference between the measured NDVI of a field and the bare soil over the difference of dense vegration's NDVI and the bare soil [8]. One concern of this estimation is defining the NDVI for bare soil and dense vegetation is difficult. Hence, several assumptions can be made. NDVI for bare soil would be assumed as the minimum NDVI for a study area over a period of time. The NDVI of dense vegetation would be the maximum NDVI of a study area under over a period of time. Implementing these assumptions in equation 3 results in equation 4.

$$FVC = \frac{NDVI - NDVI_{min}}{NDVI_{max} - NDVI_{min}} \tag{4}$$

4.3 Approach

Considering the earlier discussion, the team decided to proceed with the FVC method to estimate the full canopy dates for the fields in the dataset. There were six agricultural producers who used drones to capture images of their field. The images were captured on multiple dates between June and September of 2019. These producers were as follows:

1) Doerr
2) Kyes
3) Stech
4) Seim
5) Uhrenholdt Home
6) Uhrenholdt East

Once FVC was estimated, a histogram was produced. Figure 3 shows the FVC histogram of the Uhrenholdt Home grower's field from June 17 which was known to not have reached full canopy. Figure 4 shows the histogram of the same grower from July 29, which had reached full canopy stage. Figure 4 shows most of the values having a high FVC, which indicated the full canopy.

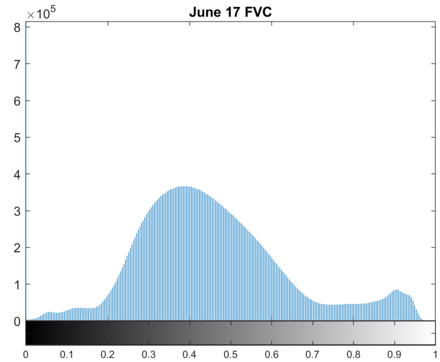

Fig. 3. This is the FVC histogram of a field that hasn't reached full canopy yet.

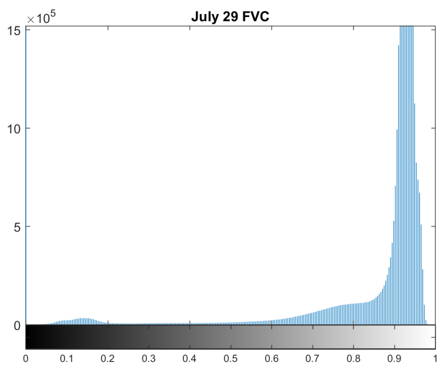

Fig. 4. This is the FVC histogram of a field that has reached the full-canopy stage already. This histogram seems to skew left, which is an indication that these plants reached later growth stages.

After plotting the FVC histogram for each image, the cumulative histogram of the FVC was created. The match distance between the two cumulative histograms were then calculated from adjacent dates [9]. It is worth noting that the match distance is a cross-bin dissimilarity measure that finds the L1 (Minkowsky-Form Distance) between two cumulative histograms. We chose the match distance because cross-bin dissimilarity measures don't suffer problems bin-by-bin dissimilarity measures encounter, such as being sensitive to histogram bin size and suffering from large frequency spikes. The cumulative histogram for the field that is not full canopy and a full canopy field are shown in figure 5 and 6 respectively.

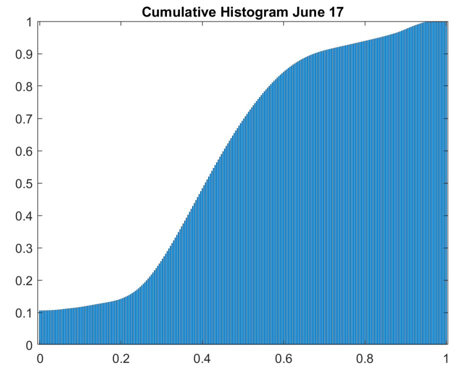

Fig. 5. This is the FVC cumulative histogram of a field that hasn't reached the full-canopy stage.

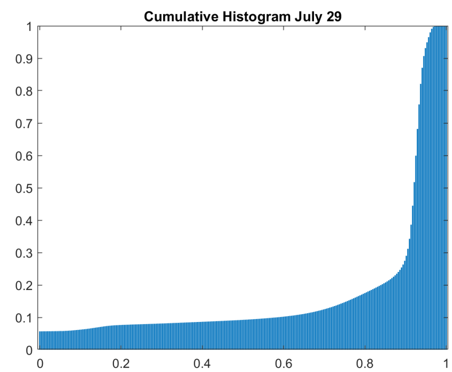

Fig. 6. This is the FVC cumulative histogram of a field that has reached the full-canopy stage.

The goal was to find when the match distance [10] was low (below 15) and plateaued. Specifically, we wanted to find when the absolute differences between two adjacent match distance measures was less than 5. The match distances are plotted in the following screeplot.

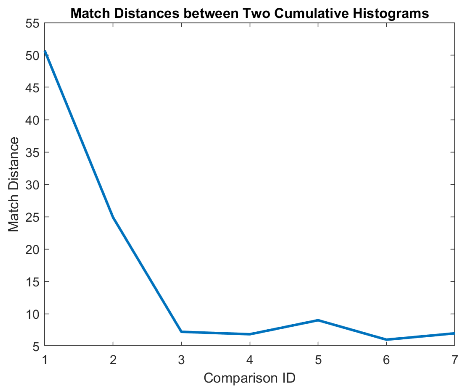

Fig. 7. This plot shows the how match distances between adjacent dates for a grower change over time. The slope of the graph is large at first, but the plot plateaus over time.

4.4 Results

Table 2 shows the results of the approach to estimate the full canopy date for the Doerr grower. As you can see, the match distance was relatively high at the beginning and plateaued by July 23rd. Therefore, the estimated full canopy date for this grower was July 23rd. At this date, the match distance was below 15 and it was close (difference of 0.4) compared to July 18th's match distance.

Table 2: This table shows the match distances between adjacent dates for the Doerr grower. The first column lists the latest date. The second column displays the match distances. The third column the change in match distance. The fourth column shows the percentage change between match distances.

Date	Match Distance	+/-	% Change
June 26 to July 2	50.67	——	——
July 11	24.89	-26	-51
July 18	7.2	-17	-71
July 23	6.82	-0.4	-5
July 29	8.99	+2	+32
August 7	5.98	-3	-34
August 13	6.96	+1	+16

Sometimes the drone images were partly obscured. This resulted in the match distance not converging. For an example, consider figure 8 and 9 below. It was found that the boundary position of the field in some of the images were not consistent.

Fig. 8. This is an image with a usual boundary.

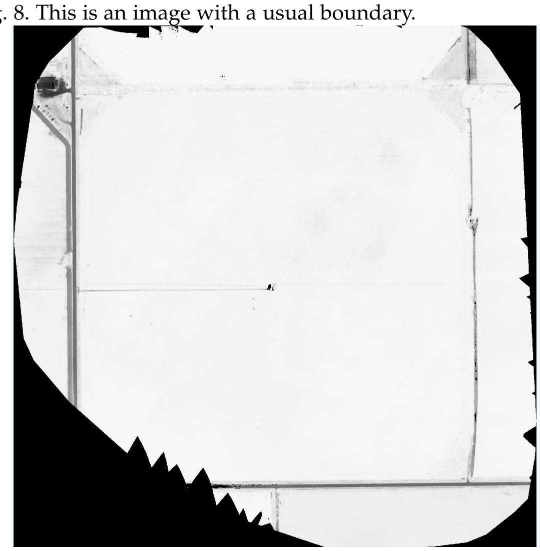

Fig. 9. This is an image from the same grower that shows an obscured boundary. This obscured boundary caused match distances to diverge.

Although the approach was found to be fruitful, some estimated full canopy dates were not as accurate as we had hoped. This issue made the Uhrenholdt East prediction suspect to skepticism. This prediction and the others are shown in table 3.

Table 3: This table lists the estimated full canopy date for each grower in the dataset.

Grower	Estimated Full Canopy Date
Doerr	July 23
Kyes	July 18
Seim	August 1
Stech	July 29
Uhrenholdt Home	July 29
Uhrenholdt East	July 16

The estimations were checked by one group member who had prior knowledge of the fields' growth logs. They validated that these estimates were reasonable.

4.5 Other Approaches

To evaluate the estimation of these results, the team attempted other methods as well. For instance, we started to perform hierarchical clustering on segments of the field. We clustered FVC intensity subimages using the match distance. Figure 10 shows an example of this method where the field resulted in the clusters shown.

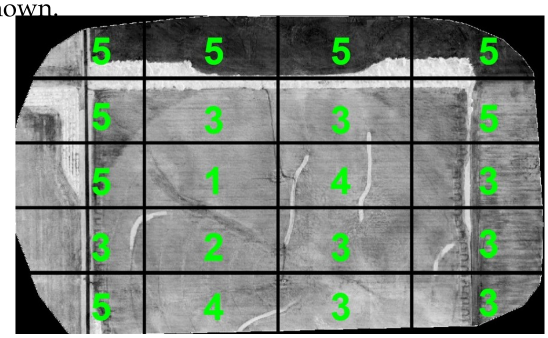

Fig. 10. This example shows the hierarchical clustering assignments for subsections of one field.

This method did not provide much information and with the limited time at hand, it was not feasible to continue developing it further. However, future work can be done to improve it as one might wonder which cluster marks a full canopy segment. To accomplish this one must make an FVC or NDVI histogram of a full canopy subimage. This histogram would be the ground truth. Then for each drone image, the subimages of each cluster could be compared to the ground truth histogram. This process is considered as supervised learning and would classify which of the clusters marks full canopy segments.

5 Detecting Clouds and Shadows

5.1 Overview

The purpose of this objective was to detect the presence of clouds and shadows within images. Detecting the presence of these objects is important for automated image processing algorithms. For researchers that perform image processing tasks by hand, it would be useful to have geospatial files for correcting image values within those regions or masking those regions out of their analysis. In order to accomplish this task, three sub-objectives were proposed:

1) Determine whether or not clouds and/or shadows are present
2) Delineate where the clouds and shadows are and find their outer boundaries.
3) Generate geospatial data files (specifically shapefiles) of cloud and shadow coverage regions.

For the development of the algorithm, we used the TerrAvion dataset because it had 12 images of acceptable quality that were verified to have clouds and/or shadows. The images could contain both clouds and shadows, just clouds, or just shadows. Though the terravion images had 8 total bands, only the first 7 were used in this algorithm because the 8th band was an alpha layer.

5.2 Related Work

Previous studies were done on the topic of shadow detection in aerial images. Some of the prior studies that inspired our approach also mentioned that reflectance properties of clouds and shadows are the most distinguishable characteristics and can be used to extract clouded and shadowed regions from images [11,12]. Our review indicated that various approaches were taken to accomplish this task. In [12], their basic process was to use two images, a brightness corrected non-shadowed image of an area and a test image. Then, they applied a wavelet transform to both images and a smoothing filter to the wavelet transformed images. Next, they thresholded the test image versus the reference image to identify areas with clouds and shadows. This step generated a binary decision mask of those regions. Using the binary decision mask, they fused the brightness corrected reference image with the test image to correct the test image in the shadowed and clouded regions. They observed that clouds, thick fog, and shadows could be detected well by their algorithm while thin fog, mist, and haze were less well detected. This is similar to what we found with thin clouds and only partial shadows.

5.3 Approach

With regards to the approach, the main code (TA_CloudDetect.m) was written without any borrowed code from available resources or references, other than function documentation shown in matlab help. Additional functions for geospatial operations were borrowed from the file exchange for implementation in functions used within the algorithm. The algorithm, as shown in figure 11, consists of four parts of Preprocessing, Cluster Detection, Delineation, and Generation/Evaluation. For your information, the algorithm could be adapted to other image formats by adjusting the preprocessing steps and some cluster detection logic and arguments; these will be addressed later.

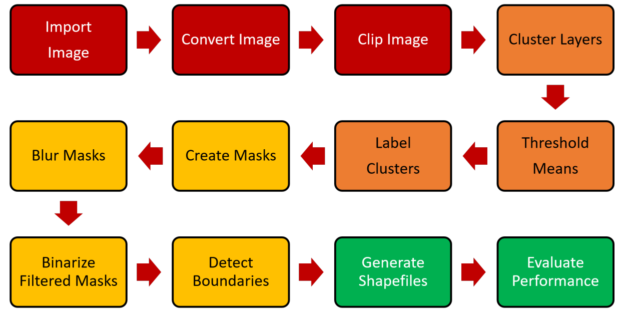

Fig. 11. Workflow for the cloud/shadow detection algorithm. The algorithm has four main sections, which are color-coated.

Several different approaches were evaluated to find clouds within images other than cluster detection; those will be addressed in the cluster detection section. The algorithm was semi-automated in the sense that except for two two user inputs for date and grower, the algorithm operated, saved image outputs, and generated excel spreadsheets along the way and sent them to the appropriate folders without intervention.

5.3.1 Preprossing

This step is illustrated in figure 12.

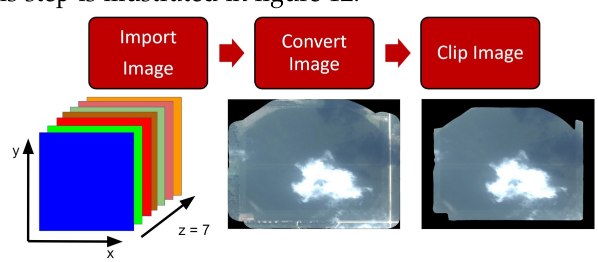

Fig. 12. The illustration lists the names and shows examples of the preprocessing step.

The preprocessing stage gets the imported images into a digestible and filtered form so that only our important regions are analyzed without extraneous noise. In order to do so, in the first step of the algorithm, terravion image is imported and its first 7 bands (Blue, Green, Red, Near Infrared, Green2, Reed2 and Thermal) and stored. Then, the terravion image was converted using the TA_ImageConverter function, which was entirely by our team. Initial images' bands, except the thermal and the alpha bands, when downloaded from the internet, were indexed in a 0 to 10,000 scale and saved as a uint16 data type. These bands were reindexed on a 0 to 1 scale and then multiplied by 255. They were rounded to convert them to unsigned 8 bit integers that can be easily used with typical functions. Thermal bands had varying concentrated ranges within the 0 to 65,535 integer range afforded by the uint16 data type. To convert the thermal band, the values in the thermal band were normalized over the range to achieve a ”thermal index” on a 0 to 1 scale that was then multiplied by 255 and rounded to convert them to unsigned 8 bit integers. This method inherently has the issue of losing the original thermal measurement values. However, this conversion worked for our purposes as relative thermal values that wouldn’t affect the comparison of thermal values in vegetation and shadows within images.

After the image layers had been converted, they were clipped to the boundaries of the field using the geospatial shapefile imported during the preprocessing and the geospatial reference object produced during the ”geotiffread” operation. After the image layers had been clipped, they were arranged to produce an NDVI indexed image. Then, the RGB layers were composited to make an RGB image. After this preprocessing, the images were deemed to be ready for cluster detection.

5.3.2 Cluster Detection

For cluster detection, there were several different potential approaches. Potential approaches were: edge detection, feature point detection, unsupervised classification, supervised learning, and deep learning. Edge detection did not work well, regardless of the edge detection method used. This failure is likely because cloud boundaries often have weak edges on the boundary of the region. Additionally, few edges within the region of the cloud as regions are fairly homogeneous. Even with attempted connection attempts and morphological operations on the detected edges, good boundary edges couldn’t be built. Feature point detection wasn’t suitable due to the fact that most feature point detection algorithms operate on square images. Fields are rarely square in the real world. When feature point detectors were applied to the entirety of the square bounded image, they only found feature points on the boundary of the image. This is because values within the square image that are not within the populated values of the image mosaic are all 0. The gradient jump is huge between 0 and a value in the mosaicked image, and those gradients exceed gradients found on the inside of the image for feature point detection. This was deemed to be ineffective. Supervised learning and deep learning were both considered, but due to the small dataset size, we decided that a machine learning approach would be suboptimal for this project.

We finally decided on unsupervised classification. This approach had a few benefits. It doesn’t need training. We also knew approximately how many clusters we were looking for so we could limit the algorithm to only looking for that many clusters. Though many different unsupervised classification approaches exist, we chose k-means for this implementation.

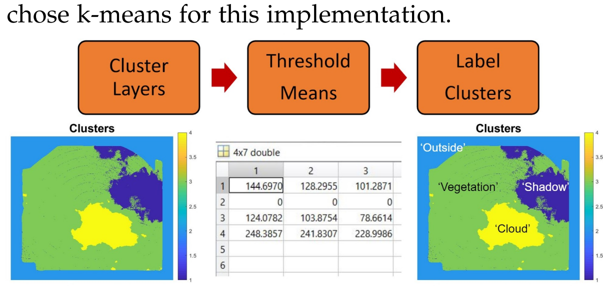

Fig. 13. This visual displays the name and example output of the cluster detection step.

The K-means algorithm used Euclidean distance with a cluster value of 4. This cluster value was based on the fact that we expected to find four basic parts of an image, which are exterior (outside of the clipped image mosaic populated pixels), clouds, shadows, and vegetation. We could have used an adaptive clustering approach and used the number of clusters for which the minimum error observed, but for an initial pass we decided to use a static approach. After k-means was carried out, cluster means were computed. Next, logic was implemented to threshold those means and label them. The logic was based on the expectation that clouds should demonstrate the most reflectance of the objects in an image while shadows should have the lowest reflectance values. Both of those regions should have lower thermal readings than vegetation. More than one vegetation cluster was allowed in the k-means clustering, while there were only one shadow, one outside, and one cloud cluster allowed. Based on this logic, a naming matrix was generated, which assigned names of cloud, shadow, outside, or vegetation to each cluster. The remaining algorithm operated off of these names to logically switch on and off operations depending on if a shadow/cloud was present.

5.3.3 Deliniation

The next step was to delineate cloud, shadow, and vegetation areas and boundaries. This process is illustrated in figure 14.

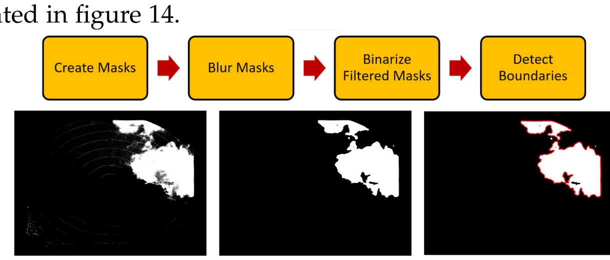

Fig. 14. This visual displays the name and example output of the deliniation step.

The first step was to create masks of these different clusters by binarizing where the clustered image was equal to the cluster value corresponding to the name of the object. These masks inherently had some noise left over from some inaccuracies in the clustering process. Therefore, these masks were blurred with a large Gaussian filter (60 by 60 with standard deviation of 60). After blurring, the mask was rebinarized such that only filtered mask values equal to one were retained in the final mask. This final mask was then used to detect boundaries of the features in the mask using the matlab function ”bwboundaries”. We imposed the algorithm so it did not include holes within the boundary process. Doing so would allow areas that could have been missed by clustering to be included in the larger bounded region anyway.

5.3.4 Generation and Evaluation

For this step, we wrote a function that drew 2D boundary shapes. The function took inputs of the boundary file, and the geospatial pixel centers (x-y dimensions) to generate a shapefile structure output and projection output. These outputs were used in conjunction with Matlab’s ”shapewrite” and ”fprintf” functions to generate shapefiles and projection files. These files were able to be opened, without projection definitions required in arcmap, and projected onto geotiff images and world imagery from ArcMap. The evaluation metrics that were chosen are the following: detection correctness, polygon overlap, centerpoint capture, and polygon similarity. Validation polygons were drawn by hand, using arcmap, and center points were also chosen by hand. Polygons were rough drawings but they covered the right general area. The evaluation process is shown in figure 15. Each step of this process is discussed further below.

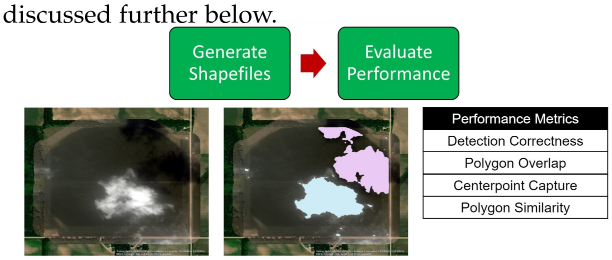

Fig. 15. This visual displays the name and example output of

the generation and evaluation step.

1) Detection correctness: did the algorithm correctly detect clouds and/or shadows? This was based on a ground-truth table that was auto loaded into Matlab at the beginning of the algorithm.
2) Polygon overlap: this metric was a yes/no decision of whether the generated shapefile polygons overlapped with the hand drawn validation polygons. Essentially, the metric tests whether or not we got the right general area.
3) Centerpoint capture: did the generated polygons contain the centerpoints that human eyes thought would be the most important to capture.
4) Polygon similarity [15]: how similar were the boundaries from the algorithm and the hand-drawn validation polygons. This metric is shown in equation 5.

$$Similarity = 1 - \frac{Area(\mathcal{R} \cup \mathcal{P}_2) - Area(\mathcal{R} \cap \mathcal{P}_2)}{Area(\mathcal{R} \cup \mathcal{P}_2)}. \quad (5)$$

The closer the area of the union and intersection of those polygons, the more similar polygons are. This measure could be greatly impacted by human error when drawing the polygons so this is the least important measure of accuracy. Nevertheless, it is still interesting to see exactly how close the algorithms boundaries got to human drawings.

5.4 Results

Overall, the algorithm demonstrated excellent performance. The outcome is shown in table 4 below.

Table 4: Evaluation metrics of the cloud and shadow detection algorithm.

	Shadows	Clouds
Detection Accuracy	92%	83%
False Negative Rate	8%	17%
False Positive Rate	0%	0%
Overlap[1]	100%	100%
Centerpoint Capture[1]	85%	94%
Polygon Similarity[1]	47%	52%
Precision	1	1
Recall	0.909	0.714
F-Score	95.2	83.3

[1] - Excluded false negatives or positives.

The algorithm correctly detected clouds and shadows in the 12 test images 83% and 92% of the time. When the algorithm was incorrect, only false negatives were observed. False negatives were 17% for clouds, 8% for shadows. When the algorithm was correct, every generated polygon file overlapped with the human-drawn polygons. Centerpoints were captured 85% of the time and 94% of the time when detecting shadows and clouds respectively. This means that the vast majority of the time the algorithm got the most important points of clouds and shadows. The generated polygons also demonstrated some similarity with the human drawn validation polygons. The similarity values were about 50% for both clouds and shadows.

Compared to results achieved by the k-means algorithm implemented by Adeline et al., our algorithm yielded better results for the detection of shadows. Adeline et al. reported an average f-score of only 85.9 across different images and numbers of groups for the k-means algorithm they implemented. Our algorithm achieved an f-score of 95.2, which significantly outperforms their algorithm for the detection of shadows.

6 Compare Shadowed Regions to the Rest of the Field

6.1 Overview

The purpose of this objective was to evaluate the impact of shadows on crop canopy measurements. We investigated how clouds and shadows influenced the nitrogen prescriptions that are calculated using vegetation indices and the sufficiency index.

6.2 Related Work

Many papers evaluated the impact of shadows on plants and how they would affect the vegetation indices that are obtained from aerial imagery. The results in [13] indicate that undetected cloud shadows can cause reflectance measurement errors of 30-40% for affected pixels, which strongly decrease the quality of NDVI measurements. Their study showed that reflectance can be used to detect and trace shadows in images. Additionally, the reflectance changes, created by shadows, strongly impact vegetation measurements. Other studies evaluated the impact of shadows as well. In [14], they found that histogram thresholding in the visible and NIR spectra was the most effective method of accounting for shadows. In their investigation of machine learning tactics, they investigated k-means and found that it had accuracies of detection within the 90% range.

6.3 Approach

To accomplish this task, the means and standard deviation of the thermal and NDVI indices were calculated for vegetated and shadowed regions of the field, as determined by the cluster detection algorithm. A two sample t-test was performed to compare the two regions. Histograms of the two indexes were plotted. It is worth noting that NDVI was used as a proxy for NDRE since they are similar indices and the NDVI can only be calculated from TerrAvion imagery.

6.4 Results

Figure 16 shows the comparison of NDVI and the themal indices in the shadowed region versus the vegetation state.

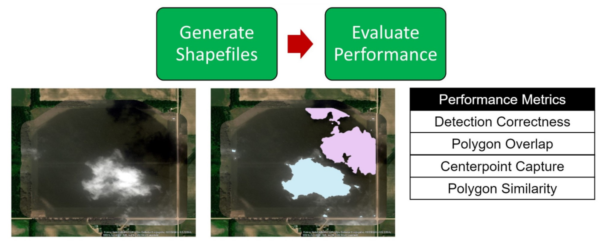

Fig. 16. These histograms compared the NDVI and the Thermal indexes between the shadowed and the vegetation regions.

As you can see below, the majority of the NDVI histogram in shadowed regions was below the histogram of the vegetated cluster. Once again, the bulk of the thermal index histogram in the shadowed region is below than from the vegetated cluster. The above results is shown in table 5 below.

Table 5: Effects of shadows on the NDVI and thermal measurements.

Parameter	NDVI	Thermal
Shadow Average	0.3732	89
Vegetation Average	0.4105	105
Standard Deviation	0.0484	15
Stat. Diff. %	100%	100%

The average NDVI for shadowed region was found to be 0.3732. The vegetation cluster had an average NDVI of 0.4105. Moreover, there was a 16 degree difference for the thermal band between the shadow and the vegetation means. These results confirmed that the presence of shadows affect the measurements of NDVI. Basic calculation showed that the obtained difference results in a sufficiency index discrepancy of 10%, which can lead to incorrectly prescribing 77 pounds of nitrogen fertilizer less than what is needed.

7 Conclusion

Cloud cover remains a challenging problem for farmers who rely on aerial imagery to maintain the health of their crops. The presence of clouds and shadows decreases the accuracy of vegetation indices, such as NDRE, that are obtained from aerial images. The farmers who use imagery these inaccuracy indices may apply suboptimal amounts of nitrogen. In some cases, they may apply more nitrogen than is necessary, which results in wasting money and harming the environment. On the other hand, they may prescribe less nitrogen, which would hurt their crops. The following paper explored ways to detect clouds on aerial images and how they would affect the NDRE measurements. To accomplish these tasks the team used the method of Full Vegetation Coverage (FVC) and estimated reasonable dates when the crop reaches full canopy. Next, the team developed an algorithm to detect clouds and shadows then trace them in aerial images. Based on the developed algorithm, clouds were successfully detected 92% of the time and shadows 83% of the time. Boundaries of shadows and clouds were found and shapefiles generated. Lastly, the clouds had impact on vegetation index measurements. This study may alarm decision-makers of the effects of shadows on aerial imagery and enable them to account for cloud and shadow effects when prescribing nitrogen, thus saving money and producing healthier crops.

8 Future Work

We foresee a few possible improvements. Some of the improvements include applying hierarchical clustering and supervised learning to detect when crops have reached full canopy. Since we have acquired histograms and can indicate when crops reach full-canopy with growth logs, we can use supervised learning to accomplish this task. Furthermore, we can further adapt the TerrAvion Cloud Detect algorithm (TA_CloudDetect.m) for detecting shadows in the drone images because these are also the images that indicate the amount of nitrogen that needs to be prescribed. Moreover, further improvements can also be done to the CloudDetect algorithm to increase the accuracy of the data. For instance, we could apply fully adaptive thresholding, adaptive clustering, or pre-cluster blurring to allow the k-means algorithms to determine the amount of clusters that minimizes the classification error. Lastly, we wish to analyze the impact of shadows on fractional-vegetation canopy measurements. Doing so can be useful since shadows can potentially result in inaccurate FVC measurements.

Acknowledgments

The authors would like to thank Dr. Ashok Samal for feedback and guidance on the objectives and methods.

References

[1] R. K., P. M. M.M and M. R. R. Varma, *Effect of cirrus cloud on normalized difference Vegetation Index (NDVI) and Aerosol Free Vegetation Index (AFRI): A study based on LANDSAT 8 images.*, 2015 Eighth International Conference on Advances in Pattern Recognition (ICAPR)., doi: 10.1109/ICAPR.2015.7050710. http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=7050710&isnumber=7050639

[2] United Nations *World population projected to reach 9.8 billion in 2050, and 11.2 billion in 2100*, UN DESA Department of Economic and Social Affairs. (n.d.)., Retrieved from https://www.un.org/development/desa/en/news/population/world-population-prospects-2017.html.

[3] Hou, L., Zheng, Y., Liu, M. et al., *Anaerobic ammonium oxidation and its contribution to nitrogen removal in China's coastal wetlands.*, Sci Rep 5, 15621 (2015), doi:10.1038/srep15621

[4] Zhang Y., Rossow W., Lacis A., Oinas V., Mishchenko M. (2004), *Calculation of radiative fluxes from the surface to top of atmosphere based on ISCCP and other global data sets: Refinements of the radiative transfer model and the input data*, Sci Rep 5, 15621 (2015), https://doi.org/10.1029/2003JD004457

[5] Fuentes, S., Palmer, A. R., Taylor, D., Zeppel, M., Whitley, R., Eamus, D. (2008)., *An automated procedure for estimating the leaf area index (LAI) of woodland ecosystems using digital imagery, MATLAB programming and its application to an examination of the relationship between remotely sensed and field measurements of LAI.*, Functional Plant Biology, 35(10), 1070-1079., https://doi.org/10.1071/FP08045

[6] Curran, Paul., *Estimating Green LA1 from MuItispectraI Aerial Photography.* , Department of Geography The University of Sheffield., https://www.asprs.org/wp-content/uploads/pers/1983journal/dec/1983_dec_1709-1720.pdf

[7] Wanjuan, S,. Xihan, M., Gaiyan R., Zhan, G., Linyuan, L., Guangjian, Y., (2017).*Estimating fractional vegetation cover and the vegetation index of bare soil and highly dense vegetation with a physically based method.*, International Journal of Applied Earth Observation and Geoinformation. Volume 58, June 2017, Pages 168-176., https://doi.org/10.1016/j.jag.2017.01.015

[8] Yu, Y., Wang, J., Liu, G. et al., *Forest Leaf Area Index Inversion Based on Landsat OLI Data in the Shangri-La City.*, J Indian Soc Remote Sens 47, 967-976 (2019) doi:10.1007/s12524-019-00950-6

[9] Rubner, Y., Tomasi, C. Guibas, L.J., *The Earth Mover's Distance as a Metric for Image Retrieval.*, International Journal of Computer Vision 40, 99-121 (2000)., doi:10.1023/A:1026543900054

[10] Schauerte, B.,*Histogram distances. Version 1.3.0.0.*, Mathworks.com., https://bit.ly/2Z3oZa1

[11] Dare, P.M., 2005.,*Shadow Analysis in High-Resolution Satellite Imagery of Urban Areas.*, Photogramm. Eng. Remote Sens. 71, 169-177.

[12] Wang, B., Ono, A., Muramatsu, K., Fujiwara, N., 1999.,*Automated Detection and Removal ofClouds and Their Shadows from Landsat TM Images.*, IEICE Trans. Inf. Syst. E82-D, 453-460.

[13] Simpson, J.J., Stitt, J.R., 1998.,*A Procedure for the Detection and Removal of Cloud Shadow from AVHRR Data over Land.*, IEEE Trans. Geosci. Remote Sens. 36, 880 -897.

[14] Adeline, K.R.M., Chen, M., Briottet, X., Pang, S.K., Paparoditis, N., 2013.,*Shadow detection in very highspatial resolution aerial images: A comparative study.*, ISPRS J. Photogramm. Remote Sens. 80, 21-38., https://doi.org/10.1016/j.isprsjprs.2013.02.003

[15] Amaral,*MathWorks, Calculate percentage of similarities in two polygons*, MATLAB Answers - MATLAB Central. MatLab Answers., https://www.mathworks.com/matlabcentral/answers/253944-calculate-percentage-of-similarities-in-two-polygons (accessed 12.17.19).