THE INTERNATIONAL SOCIETY OF
PRECISION AGRICULTURE PRESENTS THE
13th INTERNATIONAL CONFERENCE ON
PRECISION AGRICULTURE
July 31-August 4, 2016 • St. Louis, Missouri USA

Melon Classification and Segmentation Using Low Cost Remote Sensing Data Drones

Tiebiao Zhao[1], Jose M. Gonzalez[1], Jozef Franzen[1], Qi Yang[1, 2], and YangQuan Chen[1]

[1]Mechatronics, Embedded Systems and Automation (MESA) Lab, School of Engineering, University of California, Merced, 5200 North Lake Road, Merced, CA 95343, USA

[2]School of Mechanical Engineering, ShenYang LiGong University, Shenyang 110159, China

A paper from the Proceedings of the
13th International Conference on Precision Agriculture
July 31 – August 4, 2016
St. Louis, Missouri, USA

Abstract. *Object recognition represents currently one of the most developing and challenging areas of the Computer Vision. This work presents a systematic study of various relevant parameters and approaches allowing semi-automatic or automatic object detection, applied onto a study case of melons on the field to be counted. In addition it is of a cardinal interest to obtain the quantitative information about performance of the algorithm in terms of metrics the suitability whereof is determined by the final goal of the classification. Research will consist of texture analysis, color segmentation in the RGB and YCbCr color spectrums, and the combination of all extracted features. Classification methods such as manual threshold tuning and k-nearest neighbor will be used after extracting the necessary components to identify melons. Provided that the aforementioned approaches can be commonly described as feature-based, this work as aiming to cover solutions operating on both local and global scale subsequently continues by advanced techniques as for example the normalized spatial correlation based on a known sample of either texture of the whole object being sought.*

Keywords. *Object recognition, melon detection, texture analysis, color segmentation, computer vision, image processing, classification, k-nearest neighbor, data drones, agriculture, RGB, and YCbCr.*

1. Introduction

There are many different research studies in agriculture that have been published in order to increase profit and food quality. Some of which include: Weed, crop, and soil detection [3], crop yield of apple orchards [12], counting mangos [11], determining the crop coefficient of lettuce crops [4], apple tree detection [9], sprouted potatoes recognition [15], grading strawberries [7], and determining tomato quality [5]. Of which, none include research in melon detection to provide an accurate count of melons in a field.

Determining the yield in a given season is impractical due to the time involved to physically collect data in a field [8]. The cost of paying a worker to do such a time consuming task varies from the skill of the employee but the fact remains that it is inefficient. In addition, no one can guarantee that the labor will be done accurately. Keeping track of crop inventory is just not worth doing manually.

Instead, we propose using computer vision in order to keep track of how many melons are in a given field. Specifically, the concepts of automatic object recognition, classification, and segmentation will be used to complete such a task. The images used in this study will be acquired by using aerial imagery collected by low-cost remote sensing data drones. Specific details on the actual computation will be discussed later in the research.

In this research, texture analysis will be a feature that will be used to detect melons. There are many methods to extract texture features ranging from statistical methods to wavelength decomposition [13]. However, in this study we will be focusing on analyzing texture using gray level concurrence matrix to extract the needed features to identify melons in an image. The goal of texture feature extraction is to recognize the different patterns that melons can possess and try to use those recognized patterns to identify melons. Pixelwise classification has been used to determine the contrast of an image [11] so this research will try and replicate the same type of analytical concepts to identify further texture features of a melon.

We have chosen texture as one of our features because it has proven to be a good method to detect fruits when applied with edge detection [6] and even color-texture classification [1] in industrial products [2]. Combining different features will aid in creating a better object recognition algorithm to identify melons.

2. Method and Material

The study consisted of three parts: feature extraction, classification, and patch merging. Features used were color based segmentation, texture analysis, and the combination of color and texture. Each feature extracted from a library of images that consisted of both melon and non-melon images. Each feature was thus used to classify objects in a high resolution image. Manual threshold control and K-nearest neighbor classifier were used in the study to determine if an object in the image was a melon. After analyzing and classifying the whole image the third step is to merge patches to determine if the group of patches belongs to a melon.

2.1 Melon Field

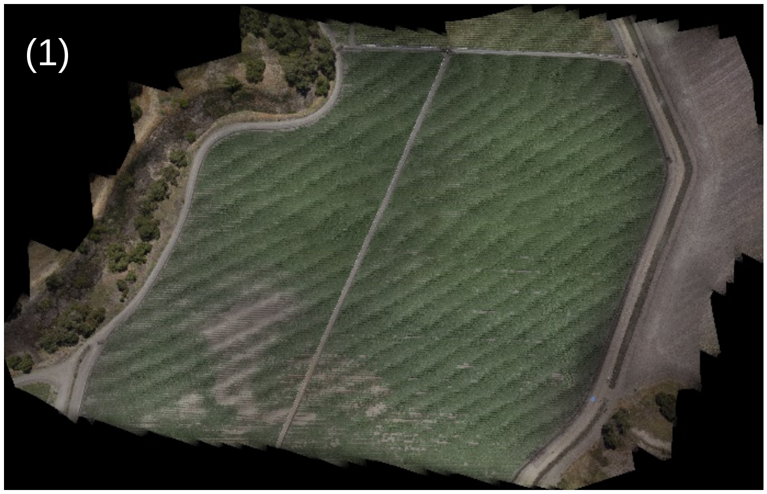

Fig 1. Melon field - Los Banos, California (37°09'47.1"N 120°47'30.7"W)

Melon field is located in Los Banos, California (37°09'47.1"N 120°47'30.7"W). Melons are grown in middle of July and harvested in middle of October, watered twice during the growing season. Images were taken two weeks before harvesting and one week after harvest. Most of the images were taken around ten in the morning. Images were captured using a DIY Quadkit (3DRobotics, Berkeley, USA) and a COTS (Commercial-off-the-shelve) camera (ELPH110HS, Canon, Japan). The ELPH110HS has a resolution of 4608 x 3459 pixels.

2.2 Library Samples

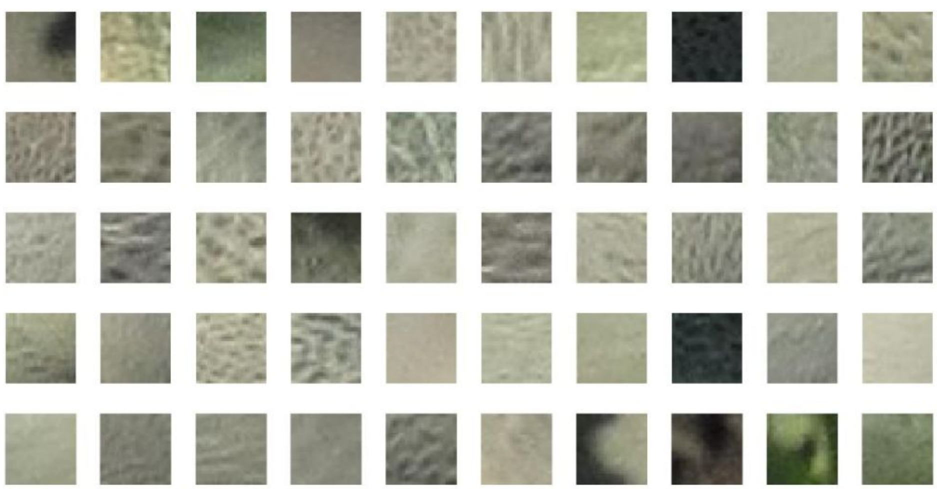

Fig 2. Melon samples used to extract the features needed to identify melons

During the research 50 melon samples were collected and used to extract the RGB, YCbCr, and texture features to identify melons. The same melon samples were used throughout the research in order to provide consistency within the results.

Fig 3. Non-melon samples used to extract the features needed to identify non-melons when using K-nearest neighbor.

During the research 50 non-melon samples were collected and used to extract the RGB, YCbCr, and texture features to identify non-melons. These samples were used when performing K-nearest neighbor classification. Details of how these samples were used are described later in the research.

2.3 Color Feature

In image processing, one of the simplest techniques to segment an image is using a color based approach. An image is essentially a matrix of values consisting of pixels in which each contains its own unique number. In color analysis, each pixel contains a numerical representation of the color it is supposed to contain in that exact space in the image-matrix. We can thus use color in order to distinguish a melon from soil and leaves because each object contains its own color pattern.

A targeted color pattern was determined using the library of melon and non-melon samples that were acquired during research. Each sample being a small portion of a melon in different images. These small samples contain the color matrix that defines a melon. Many samples were taken because of the different variation of color patterns a melon can have.

2.3.1 Color Spaces RGB and YCbCr

A color space can be thought of as a specific representation of color. The color spaces that were used in the research were RGB and YCbCr. RGB is your standard color representation of red, green, and blue color spectrum. YCbCr is a color representation with values of luminance (Y) and chrominance (CbCr).

A targeted RGB and YCbCr color was determined by using the library of melon samples and extracting the mean value contained within each of those samples. Each sample containing a certain value that could be used to identify melons in a given image.

$$Mean = \frac{\sum x_i}{n} \tag{1}$$

Mean is the average of numbers in a given data set.

2.4 Texture Feature

The second feature used in this study was texture analysis. For this feature we followed a similar method of texture analysis as in Selvarajah used in his research [14].Texture in an image can be thought of as a group of pixels containing a complex visual pattern composed of certain characteristics. Extracted characteristic can be used to distinguish different objects in an image. Texture was determined using gray-level co-occurrence matrix (GLCM) calculations within the given library of melons samples.

2.4.1 Contrast, Correlation, Energy, and Homogeneity

The four texture features that were extracted using GLCM a build in function in MATLAB were contrast, correlation, energy, and homogeneity.

$$Contrast = \sum_{i,j} |i-j|^2 p(i,j) \tag{2}$$

Contrast returns a measure of the intensity contrast between a pixel and its neighbor over the whole image.

$$Correlation = \sum_{i,j} \frac{(i-\mu i)(j-\mu j)p(i,j)}{\sigma_i \sigma_j} \tag{3}$$

Correlation returns a measure of how correlated a pixel is to its neighbor over the whole image.

$$Energy = \sum_{i,j} p(i,j)^2 \tag{4}$$

Energy returns the sum of squared elements in the GLCM.

$$Homogeneity = \sum_{i,j} \frac{p(i,j)}{1+|i-j|} \tag{5}$$

Homogeneity returns a value that measures the closeness of the distribution of elements in the GLCM to the GLCM diagonal.

2.5 Combination (Color and Texture Features)

The final approach used to identify a melon was using a combination of all the features that were extracted. Combining all the features of a melon would give a better estimation if an object in an image was actually a melon. Such a task was done by using a vector combination approach.

$$MELON = [Contrast\ Correlation\ Energy\ Homogeneity\ RGB\ YCbCr] \tag{6}$$

2.6 Classification

After extracting the features of a melon the next step is to classify an object in an image as melon or non-melon. Such a task was accomplished using two different forms of classification: manual threshold tuning and K-nearest neighbor classifier. These two forms of classification each have their own unique way of determining if an object is a melon or non-melon, however, both use the distance formula to determine how similar an object is to the features a melon possesses.

$$Distance = \sqrt{(M_1 - O_1)^2 + (M_2 - O_2)^2 + \cdots + (M_n - O_n)^2} \tag{7}$$

Distance is used to determine the similarity between known melon features ($M_{1\ldots n}$) and the features of an object ($O_{1\ldots n}$) in an image.

2.6.1 Manual Threshold Tuning

The first method used to determine if an object in an image was a melon used a simple threshold acceptance formula. Simply, if an object's distance was close to the features of a melon – the object would be classified as a melon. In this approach, the object threshold distance or similarity value would have to be determined manually through a series of trial and error to get the best result.

$$MELON < THRESHOLD < NOT\ MELON \tag{8}$$

If the distance/similarity calculation of the object were to fall below the threshold the object is considered a melon, however, the object would be considered a non-melon if the calculation were to pass the threshold.

To take it a step further, the object being considered would have to be similar to multiple melon samples, not just one sample. This increases the probability of an object to be a melon when the object is actually a melon and decreases the probability of a non-melon to be classified as a melon.

2.6.2 K-nearest neighbor classifier

The second method uses a more complex approach to classifying an object as a melon or non-melon. K-nearest neighbor classifier uses a library of melon samples and a library of non-melon samples, mainly dirt and leaves. An object's distance is calculated to both libraries and the object is classified by what library the object resembles the most (shortest distance).

The objects classification is determined by the top shortest distances from both libraries. If the object has more melon resemblance then the object is classified as a melon but if the contrary is true then the object is classified as a non-melon.

```
Algorithm K-nearest neighbor

for 1 to 10
  if sorted_list(i) == melon
    increase melon_count
  end if
end for

if melon_count > threshold
  patch is melon
end if
```

(9)

K-nearest neighbor algorithm assumes the list is already sorted from least to greatest distance. If the first 10 distances belong more to melons than to non-melon samples then the patch is very likely to be a melon.

2.7 Patch merging

Patch merging is needed after analyzing features of an image and classifying patches of the image as melons. To do such a task we iterate through the image with a set box size and merge the patches together. After merging the patches together we then analyze how many patches were merged to together and determine if that location has the enough probability to be a melon. The amount of patches needed was determined manually using a similar method as manual threshold tuning.

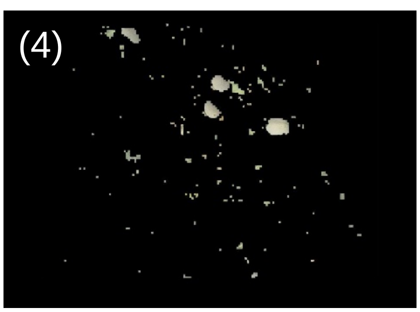

Figure 4 is a visual of how it would look like when iterating through the image. It is clear that some parts of the image contain more patches than other parts of the image. Those areas that contain a lot of patches have a high probability of a melon being at that location. Location is thus noted and the melon count is increased.

Fig. 4. RGB method using K-nearest neighbor.

3. RESULTS

Each algorithm was tested against 50 melon test samples and 50 non-melon test samples.

3.1 Manual Threshold Tuning Classification

3.1.1 RGB results

Table 1. Results from RGB color space feature approach. (Manual threshold tuning classification)

RGB	MELON (PREDICTION)	NOT-MELON (PREDICTION)
MELON (REAL)	64%	36%
NOT-MELON (REAL)	10%	90%

As shown in *Table 1*, during the test period of the research, the RGB method detected 32/50 properly (64%), 18/50 melons were not detected (36%), 5/50 non-melons were falsely detected as melons (10%), and 45/50 non-melon were detected properly as non-melons (90%).

RGB method to detecting melons gave a high 90% result on detecting non-melons properly. Later in the paper, the 90% will prove how effective the RGB method is even though it only gave a 64% chance of detecting melons.

3.1.2 YCbCr results

Table 2. Results from YCbCr color space feature approach. (Manual threshold tuning classification)

YCbCr	MELON (PREDICTION)	NOT-MELON (PREDICTION)
MELON (REAL)	62%	38%
NOT-MELON (REAL)	10%	90%

As shown in *Table 2*, during the test period of the research, the YCbCr method detected 31/50 properly (62%), 19/50 melons were not detected (38%), 5/50 non-melons were falsely detected as melons (10%), and 45/50 non-melon were detected properly as non-melons (90%).

Just like RGB, YCbCr method to detect melons gave high results in detecting non-melons properly (90%). In comparison, it detected one melon less than RGB but still a great method to detecting melons.

3.1.3 Texture Results

Table 3. Results from texture feature analysis (GLCM) approach. (Manual threshold tuning classification)

Texture (GLCM)	MELON (PREDICTION)	NOT-MELON (PREDICTION)
MELON (REAL)	52%	48%
NOT-MELON (REAL)	24%	76%

As shown in *Table 3*, during the test period of the research, the texture analysis method detected 26/50 properly (52%), 24/50 melons were not detected (48%),12/50 non-melons were falsely detected as melons (24%), and 38/50 non-melon were detected properly as non-melons (76%).

The method of detecting melons through texture analysis gave a very low result compared to the other algorithms. Later in the research, it will be proven to be the least effective due to having a 76% of correctly identifying non-melon images properly.

3.1.4 Combination Results

Table 4. Results from feature combination approach. (Manual threshold tuning classification)

Combination	MELON (PREDICTION)	NOT-MELON (PREDICTION)
MELON (REAL)	60%	40%
NOT-MELON (REAL)	10%	90%

As shown in *Table 4*, during the test period of the research, the combination method detected 30/50 properly (60%), 20/50 melons were not detected (40%), 5/50 non-melons were falsely detected as melons (10%), and 45/50 non-melon were detected properly as non-melons (90%).

Combining all the methods of detecting a melon gave similar results to the RGB feature extraction method. Combination method gave very good results despite having texture feature extraction as one of the determining factors in detecting melons, knowing that texture analysis gives low results compared to the other methods.

3.2 K-nearest neighbor Classification

3.2.1 RGB results

Table 5. Results from RGB color space feature approach. (K-nearest neighbor Classification)

RGB	MELON (PREDICTION)	NOT-MELON (PREDICTION)
MELON (REAL)	58%	42%
NOT-MELON (REAL)	2%	98%

As shown in *Table 5*, during the test period of the research, the RGB method using K-nearest neighbor classification detected 29/50 properly (58%), 21/50 melons were not detected (42%), 1/50 non-melons were falsely detected as melons (2%), and 49/50 non-melon were detected properly as non-melons (98%).

Using RGB and K-nearest neighbor classification had a significant improvement with detecting non-melons properly (98%) compared with just using RGB (90%). This 8% increase will later be demonstrate why it was such a huge improvement.

3.2.2 YCbCr results

Table 6. Results from YCbCr color space feature approach. (K-nearest neighbor Classification)

YCbCr	MELON (PREDICTION)	NOT-MELON (PREDICTION)
MELON (REAL)	60%	40%
NOT-MELON (REAL)	2%	98%

As shown in *Table 6*, during the test period of the research, the YCbCr method using K-nearest neighbor classification detected 30/50 properly (60%), 20/50 melons were not detected (40%), 1/50 non-melons were falsely detected as melons (2%), and 49/50 non-melon were detected properly as non-melons (98%).

Including K-nearest neighbor classification with YCbCr increased the rate of detecting non-melons properly, just like in RGB. However, the rate of detecting melons did not fall as great as RGB's method of detecting melons properly.

3.2.3 Texture Results

Table 7. Results from texture feature analysis (GLCM) approach. (K-nearest neighbor Classification)

Texture (GLCM)	MELON (PREDICTION)	NOT-MELON (PREDICTION)
MELON (REAL)	48%	52%
NOT-MELON (REAL)	8%	92%

As shown in *Table 7*, during the test period of the research, the texture analysis method using K-nearest neighbor classification detected 24/50 properly (48%), 26/50 melons were not detected (52%), 4/50 non-melons were falsely detected as melons (8%), and 46/50 non-melon were detected properly as non-melons (92%).

Using K-nearest neighbor classification with texture analysis gave the biggest improvement compared to the other method, with an increase from 76% to 92%. A 16% increase that caused a huge improvement with detecting melons in an image, results will be shown later in the research.

3.2.4 Combination Results

Table 8. Results from feature combination approach. (K-nearest neighbor Classification)

Combination	MELON (PREDICTION)	NOT-MELON (PREDICTION)
MELON (REAL)	60%	40%
NOT-MELON (REAL)	6%	94%

As shown in *Table 8*, during the test period of the research, the combination method using K-nearest neighbor classification detected 30/50 properly (60%), 20/50 melons were not detected (40%), 3/50 non-melons were falsely detected as melons (6%), and 47/50 non-melon were detected properly as non-melons (94%).

Using K-nearest neighbor with combination gave in improvement with detecting non-melons properly. It was noted that it did not improve as much as the rest of the methods when K-nearest neighbor was introduced, however, the improvement was noticed when melon recognition was attempted with a set images.

3.3 Melon Recognition Results (Manual Threshold Tuning Classification)

Fig. 5 – 7. Original photos taken at the melon field used to conduct the research.

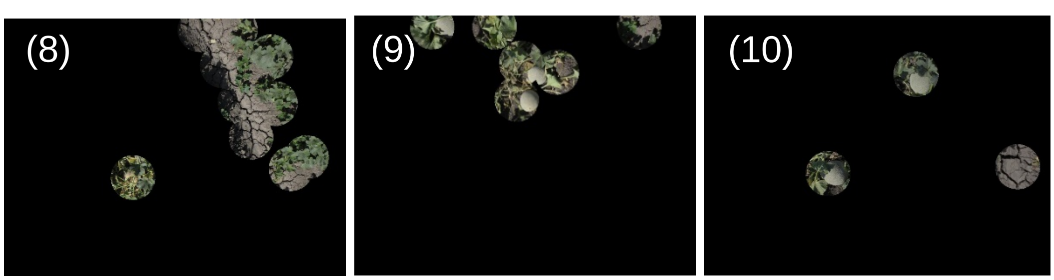

Fig. 8 – 10. Results of using the combination method to detect melons on images 5 – 7 respectively.

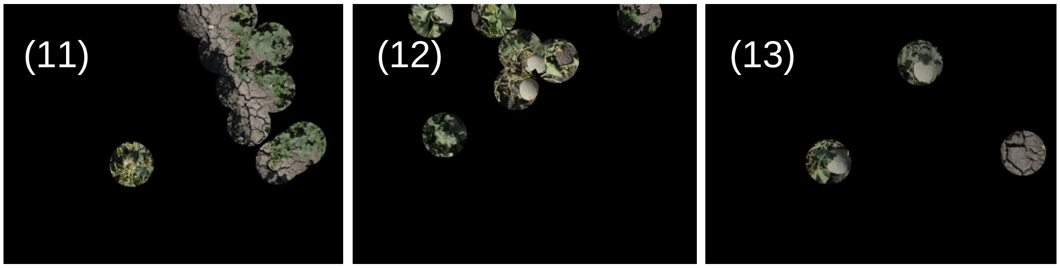

Fig. 11 – 13. Results of using the RGB method to detect melons on images 5 – 7 respectively.

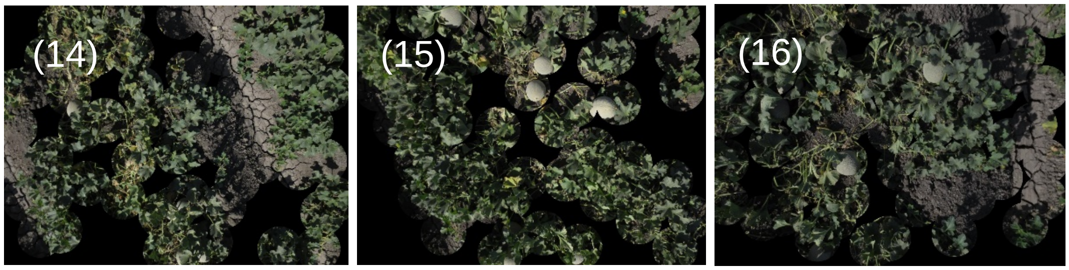

Fig. 14 – 16. Results of using the texture analysis method to detect melons on images 5 – 7 respectively.

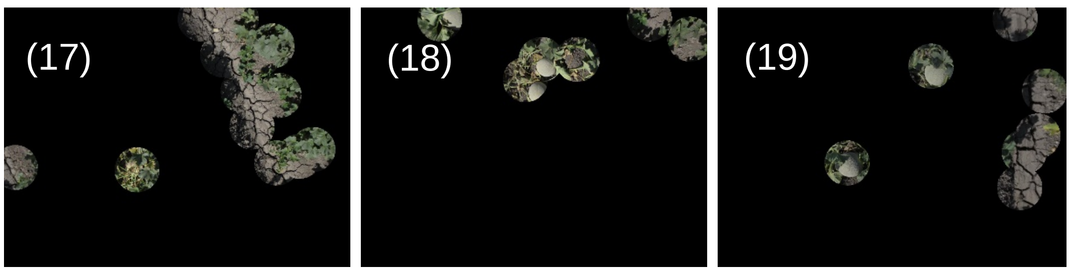

Fig. 17 – 19. Results of using the YCbCr methodto detect melons on images 5 – 7 respectively.

The results of the study using k-nearest neighbor classification can be found above. Each image highlights areas where a melon could potentially be located. It should be noted that the RGB, YCbCr, and combination methods had the best results while texture analysis fell behind. Combination had amazing results even though the method had texture features classifying melons.

3.4 Melon Recognition Results (K-nearest neighbor Classification)

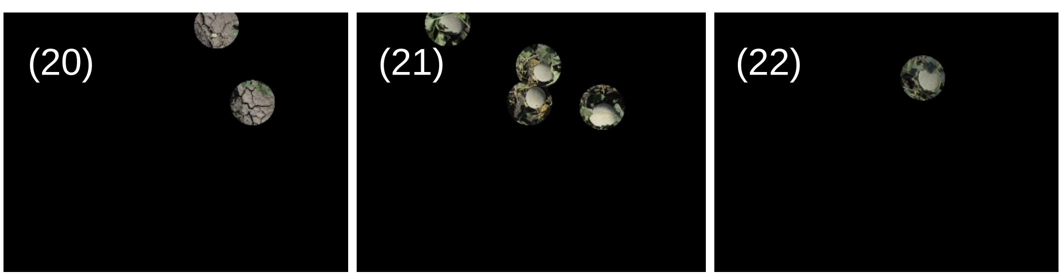

***Fig. 20 – 22.* Results of using the combination method with K-nearest neighbor to detect melons on images 5 – 7 respectively.**

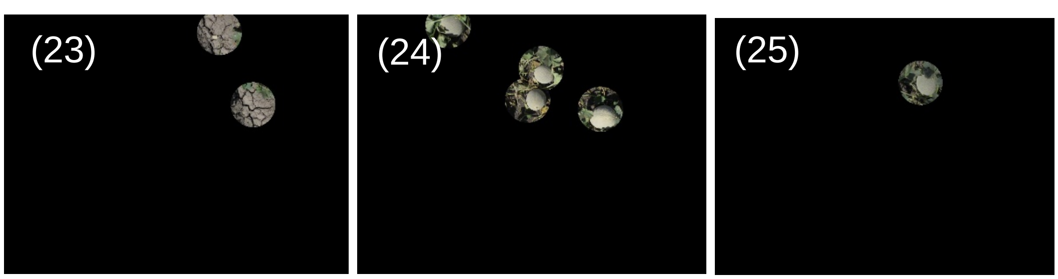

***Fig. 23 – 25*. Results of using the RGB method with K-nearest neighbor to detect melons on images 5 – 7 respectively.**

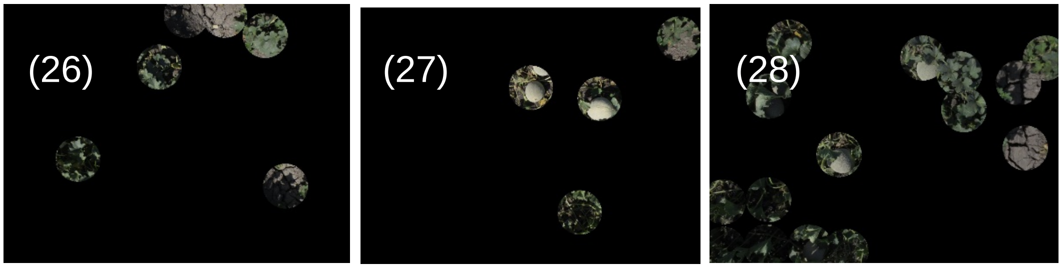

Fig. 26 – 28. R**esults of using the texture analysis method with K-nearest neighbor to detect melons on images 5 – 7 respectively.**

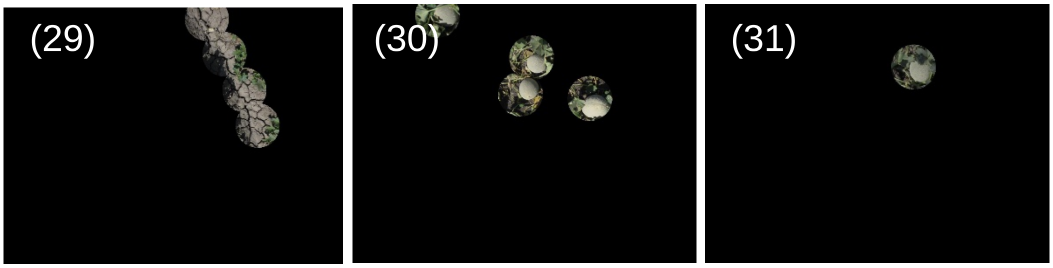

***Fig. 29 – 31.* Results of using the YCbCr method with K-nearest neighbor to detect melons on images 5 – 7 respectively.**

The results of the study using k-nearest neighbor classification can be found above. Each image highlights areas where a melon could potentially be located. Compared to manual threshold tuning method, k-nearest neighbor had better results in actually detecting melons due to a significant decrease in falsely detecting melons. When analyzing and comparing both texture results in each of the classifications, it is very apparent that k-nearest neighbor improved texture analysis by a significant amount.

4. Conclusion

Recognition, classification and localization of objects whose positions lack any regular pattern, like the analyzed study case of melons, belong to tasks completely reliant upon advanced techniques of the computer vision. Because this type of tasks can be found throughout sciences starting with self-driving vehicles and not ending with the precision agriculture, we can suppose that especially in the latter field intelligent applications of the computer vision yet await its greatest expansion. It is thus essential to conduct continuing development covering still wider variety of tasks and challenging problems to be solved by application of various fields of the AI.

We propose to use more advanced techniques to detect melons such as K-nearest neighbor due to the high results found in this study. Research can be further advanced if agglomerative merging were to be introduced. In addition, a library should be further composed of a wider variety of melons. In this study, melons found wide open were the ones to be detected. Melons that were not fully ripe or were hidden by leaves were very difficult to be recognized by the system. More research needs to be done before a wider variety of melons can be detected. In conclusion, further research is needed before an acceptable algorithm is created.

5. Acknowledgements

We are grateful to BigIdeas Project for funding to support this project and to Scott Stoddard a UCCE Merced County farm adviser for the amazing field experience on melons and for helping us find the perfect melon field to test our research. We acknowledge, undergraduate students at the University of California, Merced, Yoni Shchemelinin, Andreas Anderson, Jacob Clark for helping collect drone images and Roberto Nava, Thomas Siegle, Junguang Lu for helping prepare the image samples.

6. References

[1] Akhloufi, M. A., Larbi, W. B., & Maldague, X. (2007, October). Framework for color-texture classification in machine vision inspection of industrial products. In Systems, Man and Cybernetics, 2007. ISIC. IEEE International Conference on (pp. 1067-1071). IEEE.

[2] Akhloufi, Moulay A., Xavier Maldague, and Wael Ben Larbi. (2008). A new color-texture approach for industrial products inspection. Journal of multimedia, 44-50.

[3] Burgos-Artizzu, Xavier P., Angela Ribeiro, Alberto Tellaeche, Gonzalo Pajares, and Cesar Fernández-Quintanilla. (2010). Analysis of Natural Images Processing for the Extraction of Agricultural Elements. Image and Vision Computing, 28(1), 138–149.

[4] Fernández-Pacheco, Daniel G., David Escarabajal-Henarejos, Antonio Ruiz-Canales, Julián Conesa, and Jose M. Molina-Martínez. (2014). A Digital Image-processing-based Method for Determining the Crop Coefficient of Lettuce Crops in the Southeast of Spain. Biosystems Engineering 117, 23-34.

[5] Gastélum-Barrios, Abraham, Rafael A. Bórquez-López, Enrique Rico-García, Manuel, Toledano-Ayala and Genaro M. Soto-Zarazúa. (2011). Tomato quality evaluation with image processing: A review. African Journal of Agricultural Research, 6(14), 3333-3339.

[6] Jasani, Drashti. et al. (2015). Review of Shape and Texture Feature Extraction Techniques for Fruits. International Journal of Computer Science and Information Technologies. 6 (6), 4851 – 4854.

[7] Liming, Xu, and Zhao Yanchao. (2010). Automated Strawberry Grading System Based on Image Processing. Computers and Electronics in Agriculture 71S, S32 – S39.

[8] Maja, Jose Mari, Josue Nahun Leiva, James Robbins, and Matthew Chappell. (2015). Comparison of Manual Inventory Counts to an Automated Approach Using an Unmanned Aerial System. SNA Research Conference, 60, 72 – 77.

[9] Mao, Wenhua, Baoping Jia, Xiaochao Zhang, and Xiaoan Hub. (2009). Detection And Position Method Of Apple Tree Image. Computer and Computing Technologies in Agriculture II, 1039-1048.

[10] Ojala, Timo, and Matti Pietikäinen. (1999). Unsupervised texture segmentation using feature distributions. Pattern Recognition, 32 (3), 477-486.

[11] Payne, A.B., K.B. Walsh, P.P. Subedi, and D. Jarvis. (2013). Estimation of Mango Crop Yield Using Image Analysis – Segmentation Method. Computers and Electronics in Agriculture 91, 57-64.

[12] Qi Wang, Stephen T. Nuske, Marcel Bergerman , and Sanjiv Singh. (2012). Automated Crop Yield Estimation for Apple Orchards. 13th Internation Symposium on Experimental Robotics (ISER 2012).

[13] Ruiz, L. A., Fdez-Sarría, A., & Recio, J. A. (2004). Texture feature extraction for classification of remote sensing data using wavelet decomposition: a comparative study. International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, 35, B4.

[14] Selvarajah, S., Kodituwakku S.R., (2011). Analysis and Comparison of Texture Features for Content Based Image Retrieval. International Journal of Latest Trends in Computing, 2(1), 108–113.

[15]Tavakoli, Maryam, Mohsen Najafzadeh. (2015). Application of the Image Processing Technique for Separating Sprouted Potatoes in the Sorting Line. Journal of Applied Environmental and Biological Sciences, 4(11S), 223–227.