



American Society of
Agricultural and Biological Engineers

An ASABE Meeting Presentation

Paper Number: 073125

A Real-time Machine Vision Algorithm for Robotic Citrus Harvesting

Hannan, M.W., Postdoctoral Research Associate

Agricultural and Biological Engineering Department, University of Florida, Gainesville, FL
32611, USA

Burks, T.F., Associate Professor

Agricultural and Biological Engineering Department, University of Florida, Gainesville, FL
32611, USA

Bulanon, D.M., Postdoctoral Research Associate

Agricultural and Biological Engineering Department, University of Florida, Gainesville, FL
32611, USA

**Written for presentation at the
2007 ASABE Annual International Meeting
Sponsored by ASABE
Minneapolis Convention Center
Minneapolis, Minnesota
17 - 20 June 2007**

Abstract. Over the last several years there has been a renewed interest in the automation of harvesting of fruits and vegetables. The two major challenges in the automation of harvesting are the recognition of the fruit and its detachment from the tree. In this paper, a machine vision algorithm for the recognition of citrus fruits is presented. The algorithm consists of segmentation, region labeling, size filtering, perimeter extraction and circle detection. Evaluation of the algorithm consisted of images taken inside the canopy (varying lighting condition) and on the canopy surface. Results showed that more than 90% of the fruits were detected in the 110 images tested. In addition, the proposed segmentation was able to deal with varying lighting condition and the circle detection

The authors are solely responsible for the content of this technical presentation. The technical presentation does not necessarily reflect the official position of the American Society of Agricultural and Biological Engineers (ASABE), and its printing and distribution does not constitute an endorsement of views which may be expressed. Technical presentations are not subject to the formal peer review process by ASABE editorial committees; therefore, they are not to be presented as refereed publications. Citation of this work should state that it is from an ASABE meeting paper. EXAMPLE: Author's Last Name, Initials. 2007. Title of Presentation. ASABE Paper No. 07xxxx. St. Joseph, Mich.: ASABE. For information about securing permission to reprint or reproduce a technical presentation, please contact ASABE at rutter@asabe.org or 269-429-0300 (2950 Niles Road, St. Joseph, MI 49085-9659 USA).

method proved to be effective in detecting fruits in clusters. The development of this algorithm with its capability of detecting fruits in varying lighting condition and occlusion would enhance the overall performance of robotic fruit harvesting.

Keywords. Citrus, image processing, machine vision, robotic harvesting

Introduction

For a variety of both technological and economical reasons there has been renewed interest in the automated harvesting of fruit over the last several years. In this paper we narrow our focus to the harvesting of oranges, and more precisely we tackle the problem of orange detection. The detection of the oranges is a very important step in harvesting since the actual mechanical or robotic harvester can only harvest the oranges that have been detected. Any undetected oranges lead to a decrease in overall yield for the orange crop, and can cause an increase in costs. Proper detection of an orange requires that its location can be determined accurate enough for the robotic harvester to remove it. The detection of the oranges must be done under varying environmental conditions, and regardless of physical constraints such as leaves, branches, and unripe fruit.

The most common and intuitive approach to orange detection is through the use of machine vision (Sarig, 1993). Researchers over the previous decades have applied many different machine vision techniques for the detection of fruit. Research works on the detection of different fruits and vegetables such as apple (Grand d'Esnon, 1985; Kassay et al. 1992; Bulanon, et al. 2001; Tabb et al. 2006), cherry fruit (Tanigaki et al. 2006), cucumber (Van Henten et al, 2003), orange (Harrell et al., 1988; Hannan & Burks, 2004), tomato (Kondo et al. 1996) etc. have been reported. Early research works reported the use of monochrome cameras fitted with color filters to detect the fruits (Parrish & Goksel, 1977; Sites & Delwiche, 1985). The images were segmented based on a global thresholding approach that used data from the color filters. Several different features from the binary image such as perimeter, area, compactness, etc. were used to determine the fruits. Other researchers used morphological properties of the fruit to detect the fruit. Their algorithm looked for shape patterns such as circular arcs using edge detection, Hough transform (Whittaker et al., 1987), and circle detection (Pla, 1996). In more recent years with the advancement of sensor and computer technology, researchers have used color cameras. Harrel et al.(1988) segmented color images using a conditional probability scheme to detect oranges. Grasso and Recce (1996) used RGB thresholding to segment an image. A more in-depth review of machine vision researches in the area of fruit detection can be found in Jiminez et al (2000).

There are several problems that the previous researchers did not sufficiently solve, which can be classified into two basic types: lighting and occlusion. Lighting, as shown by previous researchers, can be a significant problem. In case of fruit harvesting, the main contributing factor to the lighting of the scene is sunlight. The amount of sunlight available is dependent on cloud cover and the incident solar angle on the scene. This can cause significant differences in how the harvesting scene appears. Furthermore, fruits inside the canopy receive a different amount of illumination compared with the fruits on the canopy surface. The image processing algorithm should be robust enough to deal with this kind of lighting variation. The second problem is that of occlusion, which minimizes the fruit area visibility and disrupts the shape of the fruit. This greatly affects the ability to detect oranges simply by their shape or size. The main causes of occlusion are leaves, branches and other fruits. Unlike leaf or branch occlusion where there is a sharp contrast in color between the fruit and leaf, fruit occlusion can cause multiple fruits to appear as a single fruit.

The inability to overcome these problems is very crucial for the success of robotic harvesting. In this paper, the development of a machine vision algorithm to detect oranges is presented. The problems of lighting and fruit occlusion are taken into account. The objective of this paper are 1) to develop an orange detection algorithm that can deal with varying lighting condition and fruit occlusion and 2) to evaluate the algorithm performance.

Materials and Methods

Development of image processing

The machine vision algorithm for orange detection consists of five image processing steps: a) segmentation, b) labeling, c) size filtering, d) perimeter extraction and e) circle detection.

a) Segmentation

Segmentation separates the object of interest (fruit) from the background (canopy). This is the first step for object recognition. Its performance is critical to fruit detection since segmentation output serves as input to succeeding processes. In this study, a global thresholding approach was proposed. The threshold was based on the chromaticity coordinate, r , which is one of the coordinates of the chromaticity diagram. The other coordinate is g . These coordinates are expressed by the following equations:

$$r = \frac{R}{R + G + B} \quad (1)$$

$$g = \frac{G}{R + G + B} \quad (2)$$

Where R , G and B are the red, green and blue color pixel values respectively.

Figure 1 shows a sample image of orange taken in the orchard. It can be observed that the r gray-level image enhanced the fruit from the background. This is also verified by the histogram, which shows a bimodal feature; the fruit distribution located at the right has the lesser number of pixels than the background.

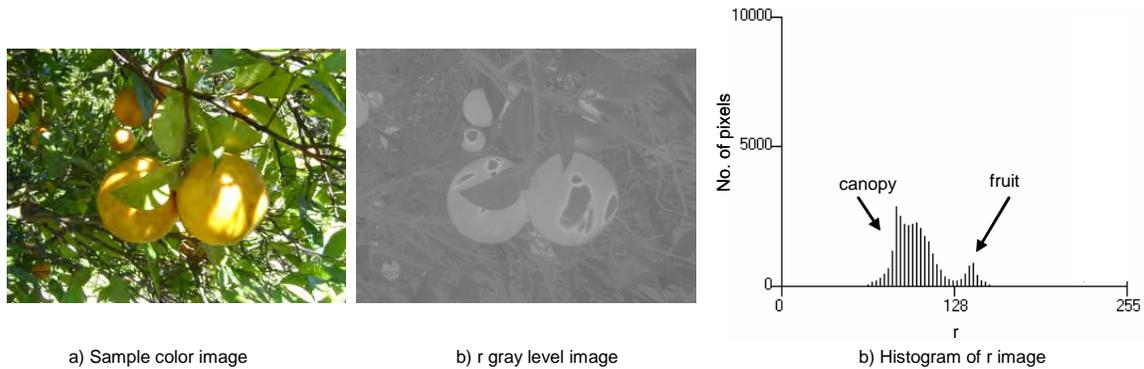


Figure 1. Sample color image with its r gray level image and histogram.

To determine the threshold value, the automatic threshold selection method used by Bulanon *et al.*(2001) was tested. The method was successful for images that have bimodal features but results in over-segmentation (objects other than the fruit is segmented) for images with unimodal feature, which are images with small areas of fruit or with no fruit present. Another method was proposed and the selection of threshold, T , is expressed by the next equation,

$$T = \mu_r + c * \sigma_r \quad (3)$$

Where μ_r = mean value of r of the image

σ_r = standard deviation of r

c = constant ranging from 1~3

Figure 2 shows an image segmented by the first approach and the second approach. The sample image contains less fruit area resulting in a unimodal histogram. The threshold calculated from the first approach is 80. The thresholded image shows that fruit were segmented but some parts of the background were also segmented as fruit. The threshold calculated from the second approach using equation 2 is 95. The fruit were segmented in the thresholded image but without the background which was segmented in the first approach.

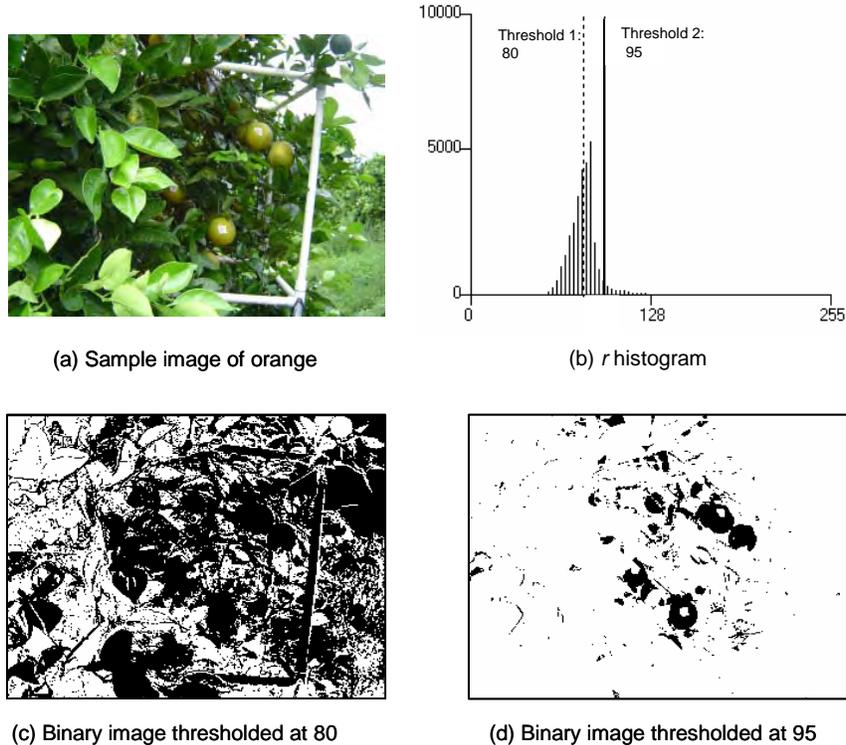


Figure 2 . Automatic thresholding using the first and second approach.

b) Labeling

Once the important color information has been extracted from the image the next step is to separate out regions of pixels in the binary image that may correspond to physical oranges. This is accomplished by applying a labeling algorithm to the binary image. The labeling algorithm searches through the image looking for pixels that are connected to each other and are defined as a region.

c) Size Filtering

After all of the regions have been found in the image, the regions must be filtered to remove any region that most likely does not belong to oranges or it is an orange but is outside the workspace of the robot. The regions of pixels can have several properties associated with them, and one important property is that of area. This property can be used to remove any region that is either too small or too large to be an orange. This is particularly useful for eliminating noise.

d) Perimeter Extraction

Perimeter extraction is the process of determining the perimeter of each labeled region. Perimeter is an important parameter to calculate geometric descriptors such as roundness, area

and etc. In this algorithm, the perimeter is used to estimate circle parameters. The process of perimeter extraction is similar to the chain coding approach wherein the contour of each region is tracked from one contour pixel to the next (Awcock, 1996). A contour pixel is an object pixel with one or more background pixel as its 4-neighbors.

e) Circle Detection (Perimeter based detection)

The circle detection method (perimeter based detection) is used to locate orange fruits in cluster. The center coordinates and the radius of the circle were calculated using a sliding segment of a predetermined number of pixels (100 pixels) that is skipped around the perimeter. Pixel coordinates of the sliding segments are used as input to calculate the circle parameters based on equation (4);

$$(x - x_c)^2 + (y - y_c)^2 = r^2 \quad (4)$$

Where (x, y) = point coordinate of circle (found on perimeter)

(x_c, y_c) = center coordinate of circle

r = radius of circle

By letting $\alpha = x_c^2 + y_c^2 - r^2$, the equation (3) can be expressed as

$$2xx_c + 2yy_c - \alpha = x^2 + y^2 \quad (5)$$

or in matrix form,

$$\begin{bmatrix} 2x_1 & 2y_1 & -1 \\ 2x_2 & 2y_2 & -1 \\ \dots & \dots & \dots \\ \dots & \dots & \dots \\ 2x_n & 2y_n & -1 \end{bmatrix} \begin{bmatrix} x_c \\ y_c \\ -\alpha \end{bmatrix} = \begin{bmatrix} x_1^2 + y_1^2 \\ x_2^2 + y_2^2 \\ \dots \\ \dots \\ x_n^2 + y_n^2 \end{bmatrix} \quad (6)$$

Equation (6) is used to estimate the circle parameters, center coordinates and radius of the sliding segment. These circle coordinates are treated as possible circle candidates and grouped together based on proximity. If the number of pixels in a center group is greater than a preset number, then the group is marked as a possible orange center. The orange center is calculated from the center group using a weighted average. The proposed algorithm was implemented using Visual C++ 6.0. A sample image processing for fruit detection is shown in Fig.3. The color image (Fig.3 (a)) was segmented using the thresholding method described above. Size filtering cleaned the segmented image from the noise. The small crosshairs in Fig. 3(e) represent the candidate circles calculated from the extracted perimeter. After clustering the candidate circles, the center of the fruit is estimated and this method successfully located the fruits.

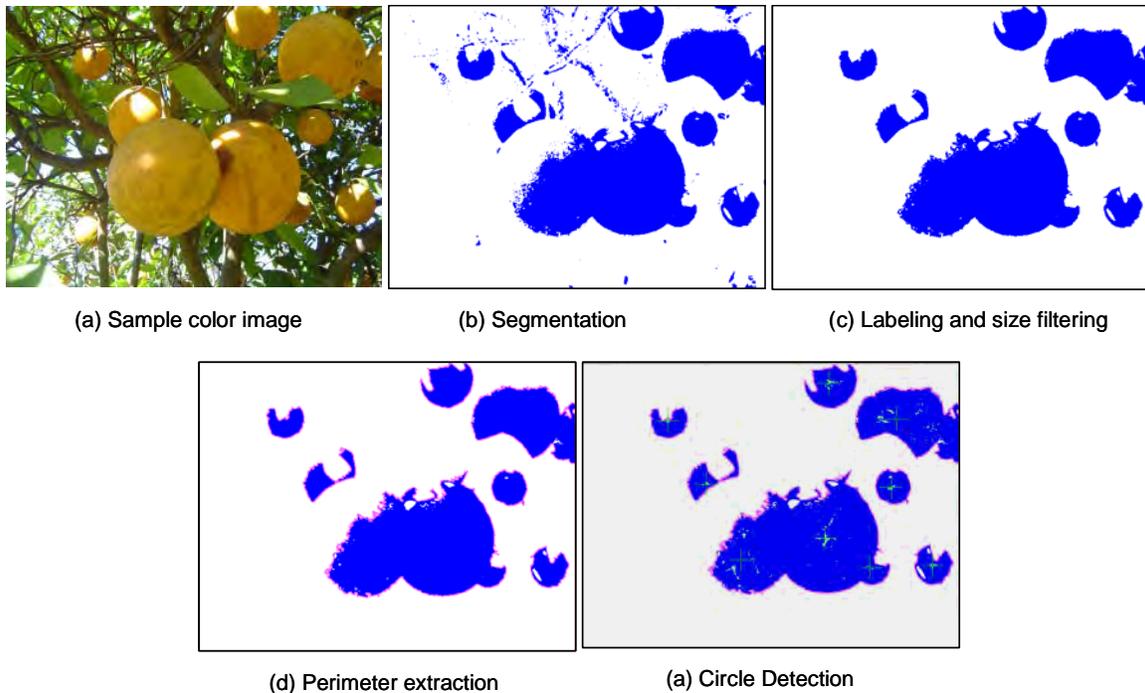


Figure 3 . Sample image processing for fruit detection.

Evaluation of Machine Vision Algorithm

To determine the capability of the algorithm, it was evaluated using three sets of images: a) inner-canopy images, b) near-view images, and c) far-view images. A digital color camera (Sony Cybershot, DSC-P8) was used to acquire the images. The images were acquired in automatic mode. Meyer et al.(2004) suggested that better classification rates can be achieved using off the shelf digital cameras used in automatic mode. The images were resampled to 640 x 480 24-bit RGB images. Lighting condition and other details are described below.

a) Inner-canopy images

Fruits inside the canopy receive different amount of illumination compared to fruits on the canopy surface. It was observed that some fruits were well-illuminated and some fruits were not. In order for the robot to harvest the fruits inside the canopy, it should be able to detect the fruits with variable illumination. To further enhance the effect of illumination on detection, the scene was acquired with flash lighting and without flash lighting. Ten different scenes were tested. For each scene, a pair of images was acquired; consisting of one for each lighting condition. A total of twenty images were acquired.

b) Near-view images

In this test, the images were acquired with the camera near the fruit (less than 0.5 m). The images were acquired under natural lighting condition. The purpose of this test is to evaluate the performance of the algorithm with respect to occlusion. Twenty five images were used for this evaluation. Since fruits in the near-view images are relatively larger in size, the size filter was set at 500 pixels.

c) Far-view images

In this test, the distance of the camera from the fruit is about 1 m. Similar to the near-view images, the algorithm is evaluated in terms of fruit detectability in the presence of occlusion. However, in far-view images, the fruits appear smaller. This will influence the segmentation process and the detection of the fruit with occlusion, especially with clustered fruits. The set of images used for this test is part of the fruit visibility study (Bulanon et al., 2007). A volume of the tree was bounded with a cube (0.5 m). Images of the six orthographic views were acquired and ten different cubes from five different trees were tested. A total of sixty images were used for this evaluation. In the size filtering operation, the filter was set at 250 pixels.

Results and Discussion

Inner canopy images

The algorithm was tested using images of fruits taken inside the canopy. Since these fruits receive varying amounts of illumination as compared with the fruits on the outer surface, the ability of the algorithm to adjust to varying lighting condition is demonstrated. Figure 4 shows a sample canopy image taken without flash (Fig.4 (a)) and with flash (Fig.4(c)) flash. The difference of luminance is noticeable in both images; the image with flash has an average luminance of 118 compared to 92 of the one without flash. However the average values of r of both images have little variation. It can also be verified in the histograms that they have similar distribution. In fact, the threshold value to segment the fruit from the canopy is nearly the same.

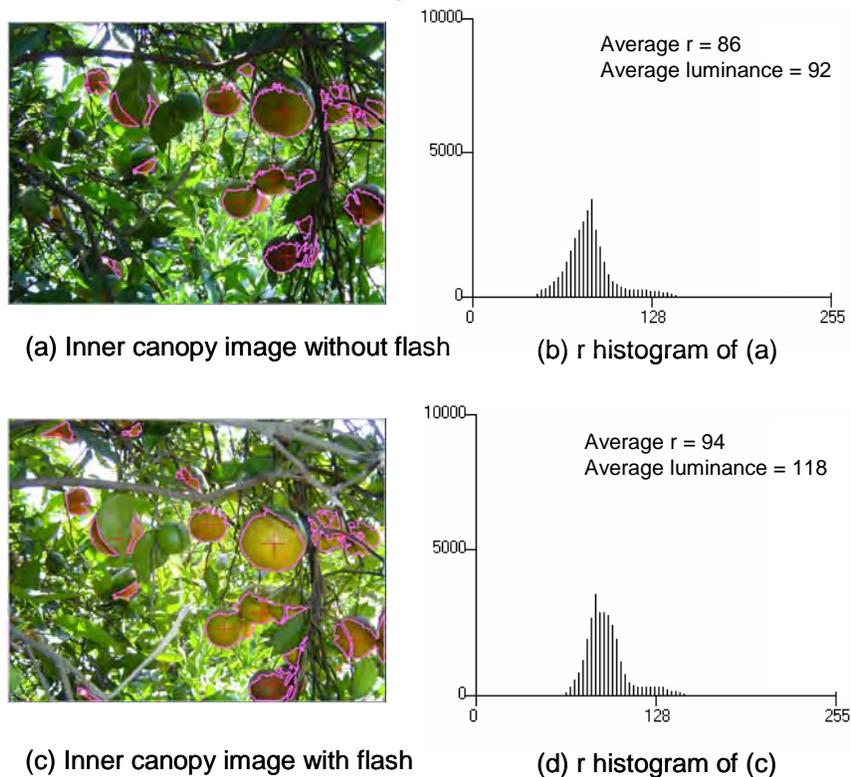


Figure 4 . Sample inner canopy images and its histogram.

Table 1 shows the result of detecting fruits inside the canopy. There were a total of 82 fruits visually counted from all the images. The algorithm detected 77 fruits in the images without artificial lighting while 81 fruits were detected with artificial lighting. It was observed that flash lighting has little influence on the recognition of fruits. This means that the proposed segmentation approach was able to adjust to the changing lighting condition. One of the strength of this approach is that the threshold is adaptively calculated from the actual image, rather than from training images.

Table 1 . Performance of fruit detection algorithm in inner canopy images.

NO. of images	No. of fruits	Without Flash		With Flash	
		Detected fruit	Detection rate %	Detected fruit	Detection Rate
20	82	77	94	81	99

Near-view images

Figure 5 shows an example of a near-view image and the result of the fruit detection using centroid based detection and perimeter based detection. Centroid based method recognized the fruit cluster composing of two fruits as single fruit while the perimeter based method detected the individual fruits in the cluster. The robot should be able to identify individual fruits in cluster to accurately grab the fruits.

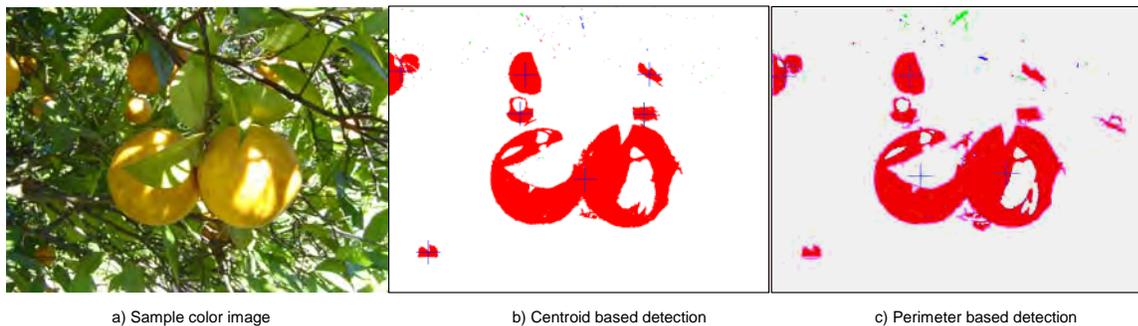


Figure 5 . Sample detection in near-view image.

Table 2 shows the result of fruit detection when the camera is near the fruit. There were 131 fruits visually counted from all the images. Fifty five of the fruits were in clusters. This is a condition of occlusion wherein a fruit is covered by another fruit. When there is no declustering method in fruit detection, fruits in cluster are detected as a single fruit. The algorithm detected 122 fruits (93%) and only 7 fruits (12%) in cluster were not detected.

Table 2 . Performance of fruit detection algorithm in near-view images.

No of images	No of fruits	No of clustered fruits	No. of detected fruits	Detection rate %	No of undetected clustered fruits	No of false detection
30	131	55	122	93	7	8

Far-view images

In the far view images, after the labeling of the segmented binary image, the fruit region was both detected using the centroid method and the perimeter based detection method. In the centroid method, the center of the region is located while circles in the region are detected in the second method. Figure 6 shows an example of far-view image detection using both methods. The cube which encloses a portion of the canopy is the region of interest. Inside the region of interest, there are six fruits identified visually. The centroid method detected four fruits while the circle detection method detected all the fruits. The circle detection method was able to detect the individual fruits in the cluster.

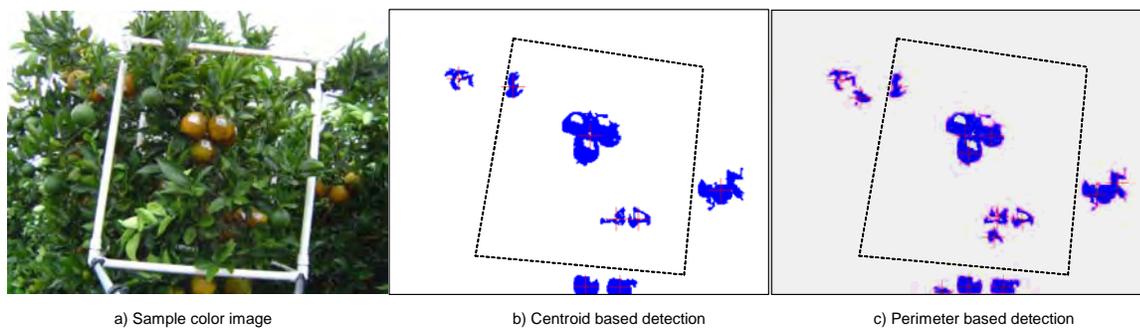


Figure 6 . Sample detection in far-view image.

Table 3 shows the result in the far-view detection. The centroid method detected 129 (59%) of the 219 fruits visually counted from all the images. The perimeter based method detected 197 fruits which improved the detection rate to 90%. This improved fruit detection was due to declustering of the segmented regions with more than one fruit. The robot should be able to recognize single fruit and clustered fruits to correctly locate the fruit and avoid damaging the fruit or its end effector. In addition, the declustering method is also useful in fruit counting for yield monitoring. This means that aside from harvesting operation, the robot could also be used for other purposes.

Table 3 . Performance of fruit detection algorithm in far-view images.

No of images	No of fruits	Fruits detected by centroid detection	Detection rate %	Fruits detected by circle detection	Detection rate %
60	219	129	59	197	90

Conclusion

A machine vision algorithm for the detection of oranges was developed and evaluated. The algorithm was composed of segmentation, labeling, size filtering, edge extraction and circle detection. It was designed to solve the problems of varying illumination and fruit occlusion through segmentation and circle detection. Segmentation used a global thresholding approach and the threshold was determined from the mean and standard deviation values of chromaticity coefficient r of the acquired image. Circle detection allowed the detection of occluded fruits. Evaluation of the algorithm consisted of images taken inside the canopy (varying lighting condition) and on the canopy surface. Results showed that more than 90% of the fruits were detected in the 110 images tested. Furthermore, the circle detection method proved to be effective in detecting fruits in clusters. The development of this algorithm with its capability of detecting fruits in varying lighting condition and occlusion would enhance the overall performance of robotic fruit harvesting.

Acknowledgements

This research was supported by Research Grant No. US-3715-05 from BARD, The United States-Israel Binational Agricultural Research and Development Fund, the Florida Department of Citrus and the Institute of Food and Agricultural Sciences at the University of Florida.

The authors would like to acknowledge the technical support of Mr.Greg Pugh and Mr.Mike Zingaro.

References

- Jiminez A.R., Ceres, R., Pons, J.L. 2000. A Survey of Computer Vision Methods for Locating Fruit on Trees, Transactions of the ASAE, 43(6), 1911-1920.
- Awcock, G., Thomas, R. 1996. Applied Image Processing, McGraw-Hill. Inc.
- Bulanon, D.M., T. Kataoka, S. Zhang, Y. Ota, and T. Hiroma. 2001. Optimal thresholding for the automatic recognition of apple fruits. ASAE Paper No. 01-3133
- Bulanon, D.M., Burks, T.F., Alchanatis, V. 2007. Study on Fruit Visibility for Robotic Harvesting. ASABE Paper No. 07-3124
- Gonzalez, R., Woods, R. 1992. Digital Image Processing. Addison-Wiley Publishing Company.
- Grand d'Esnon, A. 1985. Robotic harvesting of apples. In Proceedings of Agri-Mation 1 Conference and Exposition: 210-214. St.Joseph, MI: American Society of Agricultural Engineers.
- Hannan, M.W. and Burks, T.F. 2004. Current Developments in Automated Citrus Harvesting, American Society of Agricultural Engineers. Paper no. 043087

- Harrell, R.C., Adsit, P.D., Pool, T.A., Hoffman, R. 1988. The Florida Robotic Grove Lab. American Society of Agricultural Engineers Paper no. 88-1578.
- Jiminez A.R., Ceres, R., Pons, J.L. 2000. A Survey of Computer Vision Methods for Locating Fruit on Trees, Transactions of the ASAE, 43(6), 1911-1920.
- Kassay, L. Slaughter, DC, Molnar, S. Hungarian Robotic Apple Harvester. 1992 ASAE Summer Meeting
- Kondo, N., Nishitsuji, Y., Ling, P.P., Ting, K.C. 1996. Visual Feedback Guided Robotic Cherry Tomato Harvesting. Transactions of the ASAE, 39(6), 2331-2338.
- Parrish, E. and Goksel, K. (1977) Pictorial pattern recognition applied to fruit harvesting. Transactions of the ASAE. v20 (5): 822-827.
- Pla, F. 1996. Recognition of Partial Circular Shapes from Segmented Contours. Computer Vision and Image Understanding. 32(2), 334-343
- Plebe, A. and Grasso, G. 2001. Localization of spherical fruits for robotic harvesting, Machine Vision and Applications. 13(2):70-79
- Sarig, Y. 1993. Robotics of Fruit Harvesting: A State-of-the-art Review, J.Agr.Eng.Res. 54(4):265-280
- Sites, P.W. and Delwiche, M.J. (1985) Computer vision to locate fruit on a tree. ASAE paper# 85-3039
- Stajanko, D., Lakota, M., Hocevar, M. 2004. Estimation of number and diameter of apple fruits in an orchard during the growing season by thermal imaging, 42, 31-42
- Tabb, A., Peterson, D., Park, J. 2006. Segmentation of Apple Fruit from Video via Background Modeling. American Society of Agricultural and Biological Engineers. Paper no. 063060
- Tanigaki, K., Fujiura, T., Akase, A., Imagawa, I. 2006. Cherry Harvesting Robot, Preprints of 3rd IFAC International Workshop on Bio-robotics, Information Technology and Intelligent Control for Bioproduction Systems, September 2006, Sapporo Japan, 254-260
- Van Henten, E.J., Hemming, J., Van Tuijl, B.A.J., Kornet, J.G., Bontsema, J. Collision-free Motion Planning for a Cucumber Picking Robot. Biosystems Engineering (2003) 86(2) 135-144