A MACHINE VISION ALGORITHM COMBINING ADAPTIVE SEGMENTATION AND SHAPE ANALYSIS FOR ORANGE FRUIT DETECTION

M. W. Hannan, T. F. Burks, D. M. Bulanon

Department of Agricultural and Biological Engineering
University of Florida, Gainesville, Florida, USA
Corresponding author's e-mail: bulanon@ufl.edu.

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. This paper deals with fruit recognition and it presents the development of a machine vision algorithm for the recognition of orange fruits. The algorithm consists of segmentation, region labeling, size filtering, perimeter extraction and perimeter-based detection. In the segmentation of the fruit, the orange was enhanced by using the red chromaticity coefficient which enabled adaptive segmentation under variable outdoor illumination. The algorithm also included detection of fruits which are in clusters by using shape analysis techniques. Evaluation of the algorithm included images taken inside the canopy (varying lighting condition) and on the canopy surface. Results showed that more than 90% of the fruits visually recognized in the images were detected in the 110 images tested with a false detection rate of 4%. The proposed segmentation was able to deal with varying lighting condition and the perimeter-based 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.

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

1. INTRODUCTION

Over the past two decades, numerous researchers have studied robotic fruit harvesting. Studies on robotic harvesting of apples(Grand d'Esnon, 1985), cherries(Tanigaki et. al., 2006), oranges (Harrell, 1988; Hannan, 2004), strawberry(Nagata et al., 2006), tomatoes(Kondo et al. 1996) and others have been reported. The main reasons for these endeavors have been the shrinking labor force and the increase in labor cost, especially in the harvesting operation. Furthermore, the availability of advanced sensors and robotics technology, coupled with the increasing computational capability of computers has facilitated this development. However, up to the present, many robotic harvesting prototypes are not yet commercially available, harvesting of most fruits is still done manually. In Florida, where the main agricultural product is citrus, citrus fruit is still harvested manually. Kassler (2001) explained the reasons why a number of robotic prototypes have not proceeded to commercialization. Some of these are; a) agriculture has a work environment that is difficult to constrain to the degree achievable in a factory, b) insufficiently robust mechanical technology, c) costly mechanical technology, and d) unavailability of the basic knowledge to create technology as skillful as the human worker.

There are two approaches in automating orange harvesting; mechanical mass harvesting and the other is selective robotic harvesting. Mechanical mass harvesting (Sanders, 2005) applies some form of vibration to the tree to harvest the fruit. Because of this, there are concerns about damage inflicted to the tree, as well as the fruit. This makes mechanical mass harvesting unsuited for the fresh market. Although damage to the fruit may not be a concern for fruits destined for juice extraction, during late harvest season mechanical mass harvesting may also drop next season's fruit. Selective robotic harvesting is the other alternative, where with the proper sensors and control, can perform selective harvesting and does minimal damage both to the fruit and tree. The challenges for robotic harvesting as suggested by Sarig (1993) are the vision system for fruit recognition, the end effector for fruit removal and the coordination of these two components. The focus of this paper is on the development of the machine vision system, specifically the image processing algorithm.

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 case of oranges, one of the popular varieties of citrus, its detection is a very important step in harvesting since the actual mechanical or robotic harvester can only harvest the fruits 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, 1992; Bulanon, 2001; Tabb, 2006), cherry fruit (Tanigaki, 2006), cucumber (Van Henten et al., 2003), orange (Harrell, 1988; Hannan, 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, 1977; Sites, 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 (1997) 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 Jiménez 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 (Bulanon, 2001; Kondo, 1996: Tabb, 2006) 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 illumination 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. As described by Jiménez et al.(2000), the techniques of previous researchers to overcome these challenges were either local-based or shape-based depending on the features that they used for detection. Local-based approaches were dependent on the values associated with each image pixel to classify the pixel as a fruit or a background. Results of using local-based approaches showed that it was affected by varying lighting condition thus requiring adjustment of the threshold to adapt to the lighting condition. Slaughter (1987) attempted the use of automatic aperture control which was adjusted based on the average intensity of the fruit. However it was concluded that the problem of non-uniform illumination on the fruit was not solved.

On the other hand, occlusion which minimizes the fruit area visibility and disrupts the shape of the fruit greatly affects the detection of oranges. 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. Past researchers tried shape-based approaches which used the object's edge profile to extract the spherical feature of the fruit. However, problems of detecting spherical objects other than the fruit were encountered thus increasing false detection rate. Aside from the high false detection rate, some shape-based approach such as the Circular Hough Transform (CHT) proved to be computationally complex and inefficient (Whittaker, 1987; Pla 1996; Plebe, 2001). A combination of local-based and shaped-based methods was implemented by Jiminez et al. (2000) and Plebe and Grasso (2001). Jiménez et al.(2000) employed a laser range finder sensor and they reported that there was no false detection but only 80 % of the visible fruits were correctly detected. Although there was no false detection, the method had difficulty in detecting partially occluded fruits and it required special hardware to work in real-time. Plebe and Grasso (2001) combined a color clustering approach with an adaptive edge fitting. The color clustering used a modified Hue, Saturation and Intensity (HSI) color model to decouple the effect of intensity from the image before applying the adaptive edge fitting which proved to be more effective than the CHT. They obtained an 87% correct detection rate and a false detection rate of 15%. The failures were attributed to varying lighting condition and some types of occlusion where leaves were more of a problem than clusters of oranges.

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 visible oranges is presented. The problems of lighting variation and fruit occlusion are taken into account by employing both local-based and shape-based techniques. The objectives 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.

2. MATERIALS AND METHODS

2.1 Development of Image Processing

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

$$r = \frac{R}{R+G+B} \tag{1}$$

$$g = \frac{G}{R+G+B} \tag{2}$$

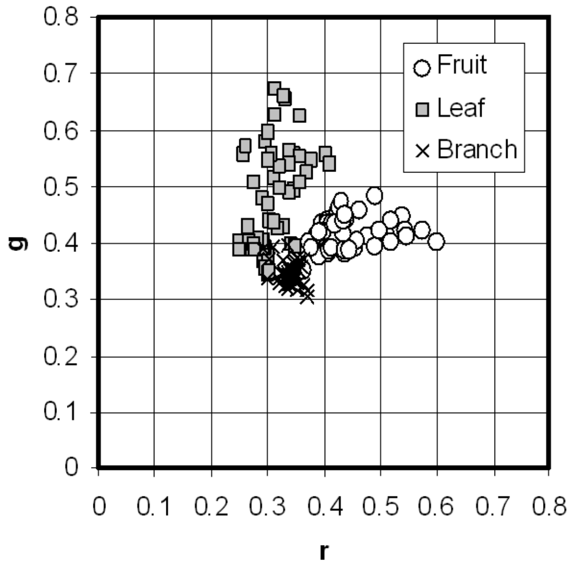
Where R, G and B are the red, green and blue color pixel values respectively.

Figure 1 shows a sample color image and the chromaticity diagram which shows the pixel distribution of the fruit, leaf and branch of the sample image. It can be observed that the fruit has the highest r value compared with the other canopy parts. Hence the chromaticity coordinate r was applied as the threshold parameter. In the subsequent processing of the images, the value of chromaticity r was normalized to 0 to 255 by multiplying equation 1 by 255.

Using chromaticity r increased the contrast of the fruit from the canopy (Fig.2) and this demonstrated its potential to segment fruit from canopy. This was also verified by the histogram, which had a bimodal feature; the fruit distribution located at the right had the lesser number of pixels than the background.

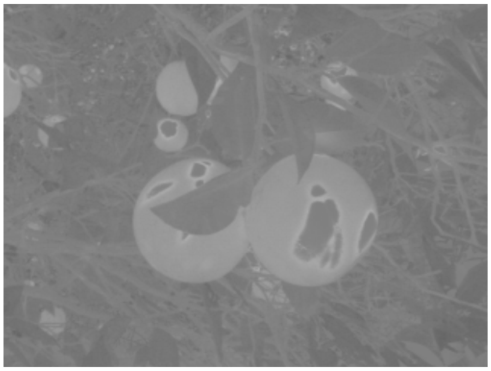
To further evaluate the classification performance of r, the Receiver Operator Characteristic (ROC) graph was employed (Fawcett, 2006). This graph is a two-dimensional graph where the true positive rate (tpr) is plotted on the y-axis and the false positive rate (fpr) is plotted on the x-axis. To evaluate r, the r gray level image was segmented using threshold values from 0 to 255. Each threshold value produced a tpr and fpr pair which corresponded to a point in the ROC graph. To calculate the total fruit pixel and the total background pixel, the fruit region was manually masked and thresholded; this was the ground truth image. The thresholded images for

a) Sample color image

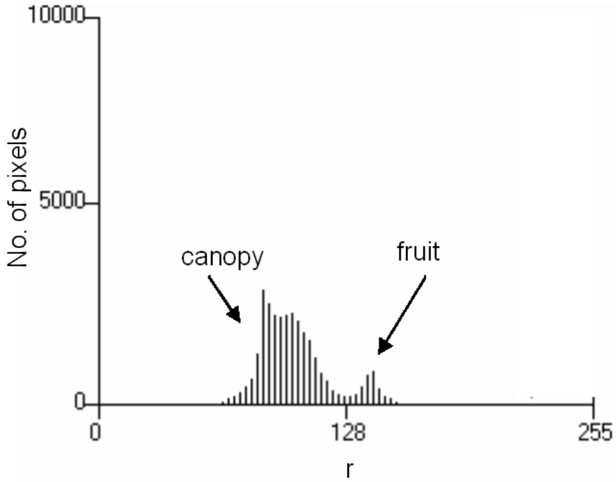


b) Chromaticity diagram

Figure 1. Sample color image of fruit in the canopy with its chromaticity diagram.



a) r gray level image



b) r histogram

Figure 2. Chromaticity r gray level image and its histogram.

the ROC calculation was compared with the ground truth image to calculate the fruit pixel correctly classified and the background pixel classified as fruit. The ROC graph of figure 2(a) shows that threshold values from 109 to 120 produced more than 0.81 tpr and less than 0.05 fpr.

$$tpr = \frac{fruit\ pixel\ correctly\ classified}{total\ \ fruit\ \ pixel} \quad (3)$$

$$fpr = \frac{background\ \ pixel\ classified\ as\ \ fruit}{total\ background\ \ pixel} \quad (4)$$

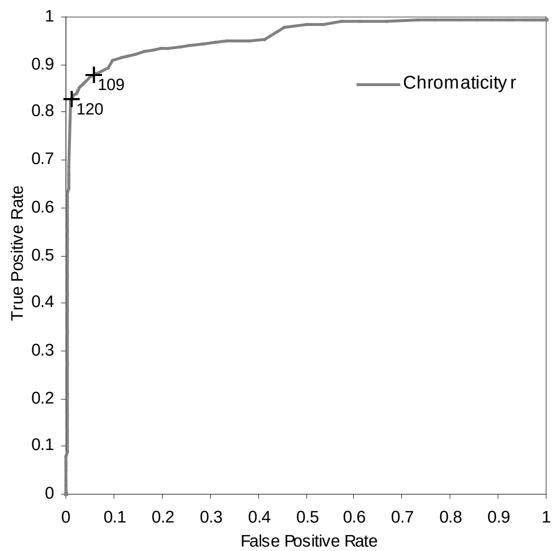


Figure 3. ROC graph of figure 2(a).

To determine the threshold value, the automatic threshold selection method used by Otsu (1979) was tested. This method calculates the threshold by locating the maximum class variance in the image histogram. It 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 to select the threshold and it was based on the ROC graph. It was found that the threshold that had high tpr and low fpr was located in the right hand side of the histogram. Using the mean value and the standard deviation of the gray level image, the threshold was calculated by the next equation.

$$T = \mu_r + c * \sigma_r \quad (5)$$

Where μ_r = mean value of r of the image

σ_r = standard deviation of r

$$c = \begin{Bmatrix} 1 & if\ \mu_r \le 90 \\ 1.3 & if\ \mu_r > 90 \end{Bmatrix}$$

The value of c was determined by trial and error. In actual robotic harvesting, the camera position varies, the area of the fruit portion in the image also varies which affects the distribution of the r histogram and thus affects segmentation, which resulted in two values for c. To evaluate if the threshold obtained in Equation 5 was optimal, the thresholds calculated from 20 test images were plotted in the ROC graph and we obtained tpr of more than 0.85 and fpr of less than 0.05.

A separate experiment conducted on Hamlin oranges investigated the effect of varying scene illuminance on chromaticity r. The region of interest (ROI) of the canopy is the area enclosed by a half meter square (Fig.4). Illuminance on the ROI was measured using a light meter and color images were acquired over a 24-hr period. Night time acquisition was aided by a portable halogen artificial lighting. Results indicated that as scene illuminance varied during the daytime, the average chromaticity r of the fruit area inside the canopy remained the same. This means that chromaticity r is robust to the varying natural lighting condition. The threshold T was also calculated using equation 5 and it varied from 116 to 138.

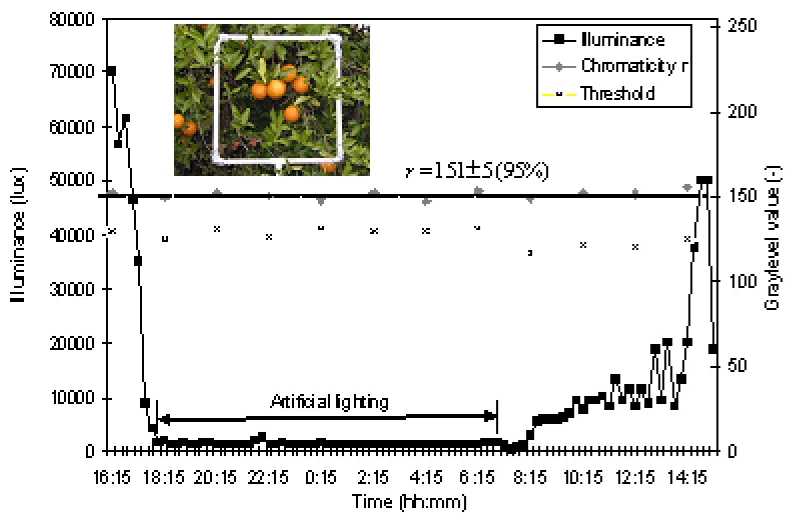


Figure 4. Effect of illuminance on chromaticity r.

b) Labeling

A labeling algorithm (Spong et al., 2006) was applied to the segmented image to separate out regions of pixels in the binary image that may correspond to physical oranges.

c) Size Filtering

The area property of the segmented region was used to remove any region that was either too small or too large to be an orange. This was 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 feature to calculate geometric descriptors such as roundness, area and etc. In this algorithm, the perimeter was 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) Perimeter-based Detection (circle detection)

The perimeter-based detection method was used to locate orange fruits in cluster. This method has two stages 1) detection of circles around the perimeter and 2) identification of fruit center (Fig.5). In the first stage, a sliding segment of predetermined number of pixels (100) was moved around the perimeter until it reached its starting point. As the sliding segment was moved, the center coordinates and the radius of the segment were calculated. Pixel coordinates of the sliding segments were used as input to calculate the circle parameters based on equation (6);

$$(x - x_c)^2 + (y - y_c)^2 = r^2 \tag{6}$$

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

(x_c, y_c) = center coordinate of circle

r = radius of circle

By letting $\alpha = x_c^{\,2} + y_c^{\,2} - r^2$, equation (6) could be expressed as

$$2xx_c + 2yy_c - \alpha = x^2 + y^2 \tag{7}$$

or in matrix form,

$$\begin{bmatrix} 2x_1 & 2y_1 & -1 \\ 2x_2 & 2y_2 & -1 \\ \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots \\ 2x_n & 2y_n & -1 \end{bmatrix} \begin{bmatrix} x_c \\ y_c \\ -\alpha \end{bmatrix} = \begin{bmatrix} x_1^2 + y_1^2 \\ x_2^2 + y_2^2 \\ \cdots \\ \cdots \\ x_n^2 + y_n^2 \end{bmatrix} \tag{8}$$

Equation 8 was solved using least square solution and it was used to estimate the circle parameters, center coordinates and radius of the sliding segment. In the second stage, these circle coordinates were treated as possible circle candidates and grouped together based on proximity. If the number of pixels in a center group was greater than a preset number, then the group was marked as a possible orange center. The orange center was 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 figure 6. The color image (Fig.6 (a)) was segmented using the thresholding method described above. Size filtering cleaned the segmented image from the noise. The small crosshairs in figure 6(e) represented the candidate circles calculated from the extracted perimeter. After clustering the candidate circles, the center of the fruit was estimated and this method successfully located the fruits.

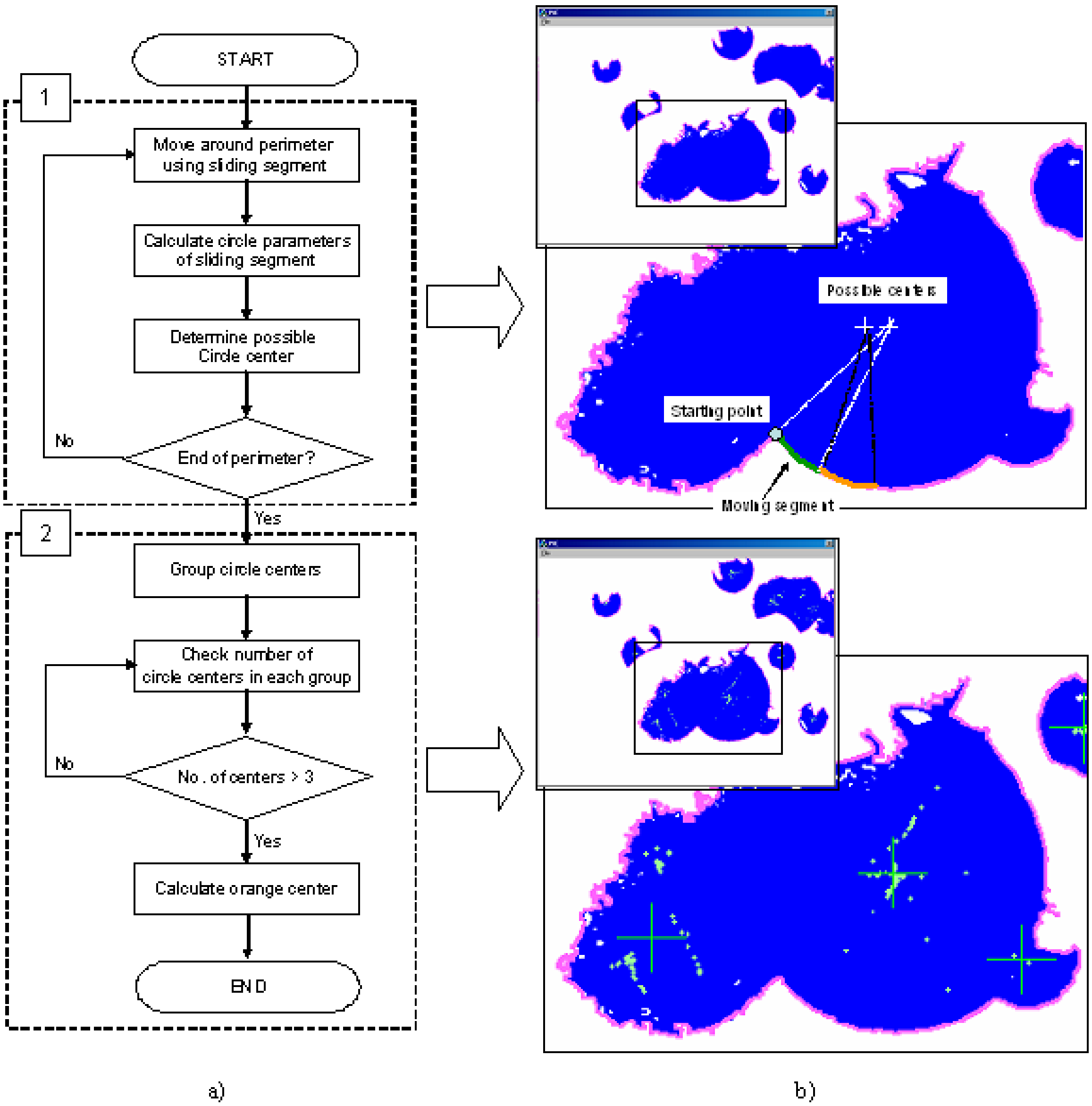


Figure 5. Perimeter-based detection showing a) flowchart and b) implementation.

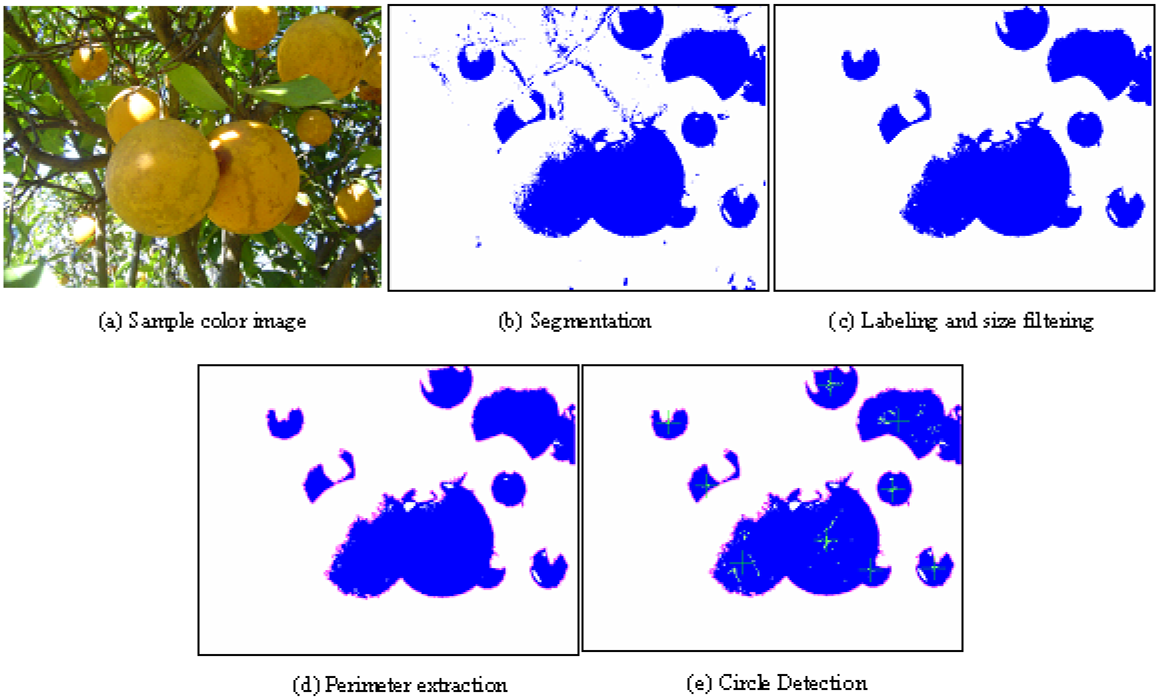


Figure 6. Sample image processing for orange fruit detection.

2.2 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 during the middle part of the day on a sunny day condition. 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 randomly 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 was to evaluate the

performance of the algorithm with respect to occlusion. Twenty five images were used for this evaluation. Since fruits in near-view images are relatively larger in size, the size filter was set to 500 pixels.
In these test images, the performance of the perimeter based detection was evaluated by comparing it with fruit detection using centroid-based detection method, which calculates the coordinates of the center of the segmented region using the zero-order moment (area of region) and the two first-order moments.

c) Far-view images
In this test, the distance of the camera from the fruit was about 1 m. Similar to the near-view images, the algorithm was evaluated in terms of fruit detectability in the presence of occlusion. However, in far-view images, the fruits appear smaller. This would influence the segmentation process and the detection of the fruit with occlusion, especially with clustered fruits. A volume of the tree was bounded with a cube (0.5 m side). 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. Similar to the near-view images, the perimeter-based detection was compared with the centroid-based detection.

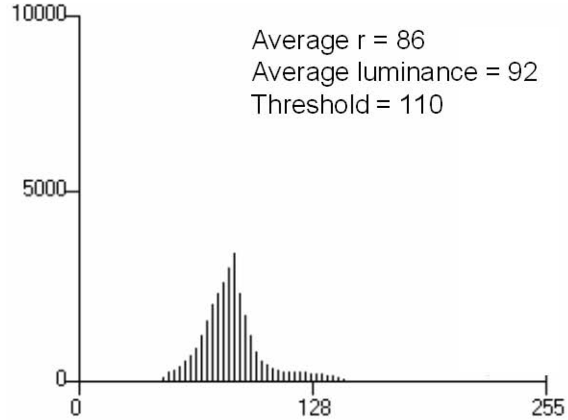
In the size filtering operation, the filter was set to 250 pixels. The estimated distance from camera to the scene was the basis for determining the size filters in both the near-view and far-view images. In actual robotic harvesting, the distance from camera to the scene is unknown, therefore setting the size filter may be difficult. However an estimated distance can be provided by another instrument such as an ultrasonic sensor.

3. RESULTS AND DISCUSSION

3.1 Inner Canopy Images

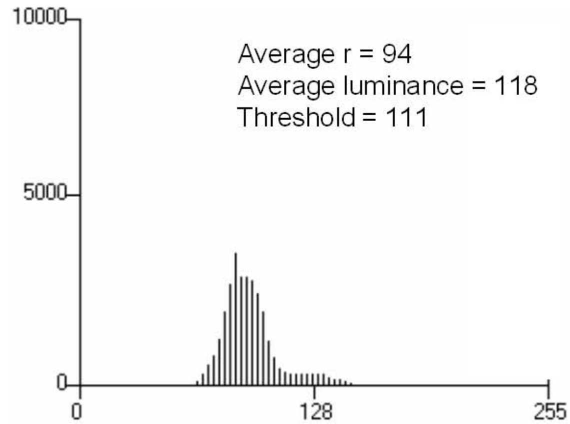
Fruits in the inner canopy receive varying amount of illumination as compared with fruits on the outer surface because the dense canopy and the branches block some of the light penetrating the canopy. In this test, the ability of the algorithm to detect the fruits in the inner canopy was demonstrated. The acquisition of images with flash (Fig.7(a)) and without flash (Fig.7(c)) enhanced the effect lighting variation. The difference of luminance was noticeable in both images; the image with flash had an average luminance of 118 compared to 92 of the one without flash. However the average values of r of both images had minimal variation and their histograms had similar distribution. In fact, the threshold value to segment the fruit from the canopy was almost equal.

(a) Inner canopy image without flash



(b) r histogram of (a)

(c) Inner canopy image with flash



(d) r histogram of (c)

Figure 7. Sample inner canopy images taken with flash and without flash and their respective histograms.

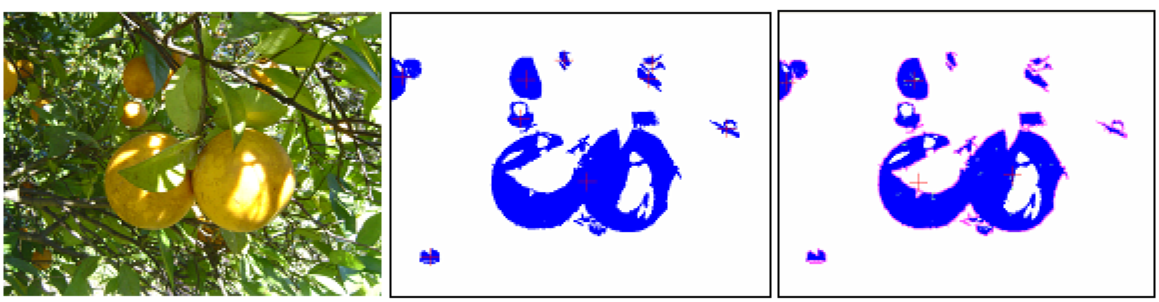
There were a total of 82 fruits visually counted from the twenty images tested (Table 1). In the images taken without flash, 94% of the visible fruits were detected with 1.2% false detection rate, while 99% of the fruits were detected in the images with flash with 3.7% false detection rate. The false detections were leaves that were brightly lit with flash. High detection rates and low false detection rates were obtained in both conditions. A paired t-test conducted on the detection results of the flash and without flash conditions resulted in $P > 0.05$ which indicated that flash lighting had insignificant influence on fruit recognition. This was attributed to the thresholding approach which was able to adapt 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. One disadvantage of using trained classifiers is the dependency on the training images. Slaughter (1987) suggested that a statistical classifier should be retrained whenever the appearance of the objects in the scene change to achieve optimal performance in segmentation.

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

		Without Flash			With Flash		
No. of images	No. of fruits	Detected Fruit	Detection rate, %	False Detection	Detected Fruit	Detection rate, %	False Detection
20	82	77	94	1	81	99	3

3.2 Near-view Images

To illustrate the performance of the centroid-based detection method and the perimeter-based detection method, a sample image (Fig.8) containing a fruit cluster with two fruits was tested. The centroid-based detection method recognized the fruit cluster as a single fruit while the perimeter-based detection method detected the individual fruits in the cluster. The reason for this was that during segmentation, the fruit cluster was segmented as a single blob. In the centroid-based method, the center of the blob was calculated while the perimeter-based method searched for the number of possible circles in the blob which resulted in the detection of multiple fruits. The robot should be able to identify individual fruits in cluster to accurately grab the fruits. In addition, if this algorithm would be used in orchard yield monitoring (Annamalai and Lee, 2003), detecting individual fruits in clusters could provide a better yield count.



a) Sample color image b) Centroid based detection c) Perimeter based detection

Figure 8. Sample near-view image with fruits detected using centroid based detection and perimeter based detection.

There were 131 fruits visually counted from the 30 near-view images (Table 2). Fifty five of the fruits were in clusters. This was a condition of occlusion wherein a fruit was covered by another fruit. As earlier mentioned, when there is no declustering method, fruits in cluster are detected as a single fruit. The algorithm detected 122 fruits (93%) and only 7 fruits (5%) in cluster were not detected. The performance of the perimeter-based detection method was mainly affected by the visible area of the fruit. This caused some of the fruits in cluster not to be counted. Similar to previous studies, some brightly-lit leaves and branch were detected as fruit (6%). This can be observed in figure 8 where a brightly-lit leaf and branch were segmented as fruit. The main reason for this was the high degree of variability of the scene. Figure1 demonstrated this variability where there was an overlap between some leaf pixels and fruit pixels. This could be resolved by increasing c in equation 5 and increasing area filter size. Increasing c decreases the amount of overlap but it also decreases the segmented fruit portion. While increasing area filter removes these falsely detected areas, it may also removed correctly segmented areas. These values that control the performance of the algorithm should be tuned to achieve a high fruit detection rate and a low false detection rate. Compared to past results, improvement in both fruit detection rate and false detection rate were obtained using this algorithm.

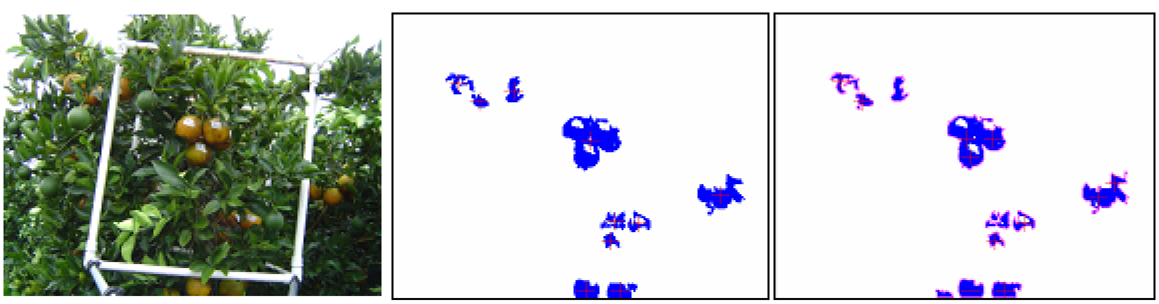
Table 2. Performance of fruit detection algorithm with near-view images.

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	9

3.3 Far-viewImages

Compared to near view images, fruits in the far view images appear smaller in size. In actual robotic harvesting, the position of the camera relative to the canopy will be varying. As the camera position varies, the area of the fruit portion in the image also varies which affects the distribution of the r histogram and thus affects segmentation. This is the reason for the two values of c in equation 5. The robot should be able to deal with the effect of the camera position variation.

The sample color image shows a region of interest bounded by a cube (Fig.9(a)). Six fruits were identified inside the region of interest and three of them were in a cluster. The centroid-based method (Fig.9(b)) detected four fruits while the perimeter-based (Fig.9(c)) method detected all the fruits including the fruits that were in cluster. The other effect of varying camera position is the size variation of the fruits in the image. The results of the near-view and far-view tests showed that the perimeter-based detection was able to detect the fruits from different camera positions.



a) Sample color image b) Centroid based detection c) Perimeter based detection

Figure 9. Sample far-view image with fruits detected using centroid based detection and perimeter based detection.

In all the images that were tested, the centroids-based method detected 129 (59%) of the 219 fruits visually counted from all the images (Table 3). The perimeter-based method detected 197 fruits which improved the detection rate to 90%. This improved fruit detection was brought about by the declustering of the segmented regions with more than one fruit. Correct recognition of fruits in cluster increases harvesting accuracy and it also provides protection against damage to the fruit or the robotic hand.

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

No. of images	No. of fruits	Fruits detected by centroid detection	Detection rate, %	Fruits detected by perimeter-based	Detection rate, %	False detection
60	219	129	59	197	90	8

Overall, the fruit detection algorithm demonstrated its detection capability with a correct detection rate of 93% and a false detection rate of 4% using a total of 110 test images. The combination of both local-based (r thresholding) and shape-based (perimeter detection) proved to be effective in detecting the oranges. Although the results still showed evidence of the influence of lighting variation on segmentation similar to what Plebe and Grasso (2001) had reported, it had a lower false detection rate and a higher fruit detection rate compared to what they had obtained. While Jiminez et al. (2000) approach showed its robustness by not getting false detection, it was still affected by occlusion and they commented that it requires special hardware to work in real-time.

Future works would involve the employment of the machine vision algorithm to control a robotic harvesting manipulator, evaluate its vision-control performance, and to improve the algorithm with the goal of lowering false detection.

4. CONCLUSIONS

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 perimeter-based detection. It was designed to solve the problems of varying illumination and fruit occlusion through segmentation and perimeter-based detection. The following are the conclusions derived from this study:

a) Thresholding using chromaticity r demonstrated its potential in dealing with varying lighting condition. The main reason for this is that chromaticity r is a normalized color value in which the intensity component is decoupled while retaining the hue property.

b) The perimeter-based detection showed that it has the potential to separate fruits in cluster. Analysis of the perimeter using a sliding segment to determine the number of fruits in a segmented blob was effective in dealing with blobs of different sizes.

a) The combination of r segmentation (local-based approach) and perimeter-based detection (shape-based approach) proved to be effective in detecting the orange fruits. Results showed a success rate of 93 % and a false detection rate of 4 %. Future work would involve the integration of the machine vision algorithm to actual robotic harvesting.

5. 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.

6. REFERENCES

Annamalai, P., Lee, W.S. 2003. *Citrus Yield Mapping System Using Machine Vision*. ASAE Paper No.031002. St. Joseph, Mich.:ASAE.

Awcock, G., Thomas, R. 1996. *Applied Image Processing*, McGraw-Hill. Inc.

Bulanon, D.M., Burks, T.F., Alchanatis, V. 2007. *Study on Fruit Visibility for Robotic Harvesting*. ASABE Paper No. 07-3124. St.Joseph, Mich.:ASABE.

Fawcett, T. 2006. *An introduction to ROC analysis*. Patterm Recognition Letters 27(2006) 861-874.

González, 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: ASAE.

Grasso, G. and Recce, M. 1996. *Scene analysis for an orange harvesting robot*. Artif Intell Appl 11: 9-15.

Hannan, M.W. and Burks, T.F. 2004. *Current Developments in Automated Citrus Harvesting*, ASAE Paper no**.** 043087. St.Joseph, Mich.:ASAE.

Harrell, R.C., Adsit, P.D., Pool, T.A., Hoffman, R. 1988. *The Florida Robotic Grove Lab*. ASAE Paper no**.** 88-1578. St.Joseph, Mich.:ASAE.

Jiménez A.R., Ceres, R., Pons, J.L. 2000. *A Survey of Computer Vision Methods for Locating Fruit on Trees*, Trans. ASAE 43(6): 1911-1920.

Kassay, L. Slaughter, DC, Molnar, S. 1992. *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*. Trans. ASAE 39(6): 2331-2338.

Otsu, N. 1979. *A threshold selection method from gray - level histograms*. IEEE transactions on Systems, Man, and Cybernetics. SMC-9(1): 62-66.

Parrish, E. and Goksel, K. 1977. *Pictorial pattern recognition applied to fruit harvesting*. Trans. ASAE 20 (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: 70-79.

Sanders K.F. 2005. *Orange Harvesting Systems Review*. Biosystems Engineering 90(2): 115-125.

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 no. 85-3039. St.Joseph, Mich.:ASAE.

Slaughter D.C. 1987. *Color Vision for Robotic Orange Harvesting*. PhD diss. Gainesville, Florida: University of Florida, Department of Agricultural Engineering.

Spong, M.W., Hutchinson, S., Vidyasagar, M. 2006. *Robot Modeling and Control*. John Wiley and Sons, Inc.

Tabb, A., Peterson, D., Park, J. 2006. *Segmentation of Apple Fruit from Video via Background Modeling*. ASAE Paper no. 063060. St.Joseph, Mich.:ASAE.

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. 2003. *Collision-free Motion Planning for a Cucumber Picking Robot*. Biosystems Engineering 86(2): 135-144.

Whittaker, A.D., Miles, G.E., Mitchell, O.R., Gaultney, L.D. 1987. *Fruit location in a partially occluded image*. Trans. ASAE 30(3): 591-596.