World Academy of Science, Engineering and Technology
International Journal of Computer and Information Engineering
Vol:11, No:1, 2017

Detecting Tomato Flowers in Greenhouses Using Computer Vision

Dor Oppenheim, Yael Edan, Guy Shani

Abstract**—**This paper presents an image analysis algorithm to detect and count yellow tomato flowers in a greenhouse with uneven illumination conditions, complex growth conditions and different flower sizes. The algorithm is designed to be employed on a drone that flies in greenhouses to accomplish several tasks such as pollination and yield estimation. Detecting the flowers can provide useful information for the farmer, such as the number of flowers in a row, and the number of flowers that were pollinated since the last visit to the row. The developed algorithm is designed to handle the real world difficulties in a greenhouse which include varying lighting conditions, shadowing, and occlusion, while considering the computational limitations of the simple processor in the drone. The algorithm identifies flowers using an adaptive global threshold, segmentation over the HSV color space, and morphological cues. The adaptive threshold divides the images into darker and lighter images. Then, segmentation on the hue, saturation and volume is performed accordingly, and classification is done according to size and location of the flowers. 1069 images of greenhouse tomato flowers were acquired in a commercial greenhouse in Israel, using two different RGB Cameras – an LG G4 smartphone and a Canon PowerShot A590. The images were acquired from multiple angles and distances and were sampled manually at various periods along the day to obtain varying lighting conditions. Ground truth was created by manually tagging approximately 25,000 individual flowers in the images. Sensitivity analyses on the acquisition angle of the images, periods throughout the day, different cameras and thresholding types were performed. Precision, recall and their derived F1 score were calculated. Results indicate better performance for the view angle facing the flowers than any other angle. Acquiring images in the afternoon resulted with the best precision and recall results. Applying a global adaptive threshold improved the median F1 score by 3%. Results showed no difference between the two cameras used. Using hue values of 0.12-0.18 in the segmentation process provided the best results in precision and recall, and the best F1 score. The precision and recall average for all the images when using these values was 74% and 75% respectively with an F1 score of 0.73. Further analysis showed a 5% increase in precision and recall when analyzing images acquired in the afternoon and from the front viewpoint.

Keywords**—**Agricultural engineering, computer vision, image processing, flower detection.

I. Introduction

DETECTING objects using computer vision in field conditions is a key requirement for automating and improving many tasks in agriculture. Harvest of fruits and vegetables, pest control, pollination and yield estimation are only some of these potential tasks. However, without accurate and fast detection, these tasks could not compete with human labor. In recent years many researches have been dealing with the challenging task with limited success, mainly because of the diverse and complex agricultural environment [1].

Pollination as an example, is performed today mainly by bees. However, bee population has been suffering from colony collapse disorder in recent years, which have reduced the number of bees available for pollination [2]. As a result, prices of many agricultural products could rise [3], [4]. To address these ecological and economic problems, researchers have proposed the use of mini unmanned aerial vehicle (UAV) as a solution to the pollination problem. In a Harvard project, researchers are trying to develop a bee size UAV called the Robobee [5]. As the main part of Robobee's navigation system a vision system is developed, which uses image sensors in order to provide information of the Robobee's close proximity.

Robotic drone pollinators such as the Robobee face several challenges in a greenhouse. The drone should be able to navigate within the rows of the plants and avoid damage to the plants, while performing its main task of detecting the flowers suitable for pollination and pollinating them. Detecting the flowers using computer vision is a complex task, and despite the progress in computer vision and image processing technologies, applications for agricultural use in the field have been scarce [1]. The main reason is the complex unstructured and cluttered agricultural environment highly variable lighting conditions and target's physical features and occlusions of the targeted object by other fruits or foliage [6].

The techniques used in target detection can be divided into two main groups according to the features they use – Local based techniques and shape based techniques. Local based techniques use the values of each single pixel to decide whether the pixel belongs to the target or the background and tend to be faster and easier for implementation [7]. Shape based features examine a group of pixels and their relations to each other. Shape features rely on the fact that most targets in agricultural environment have distinct shape in comparison to their background and are more invariant to lighting conditions. However, shape features are not flawless, and they tend to be more computationally demanding [8]. The use of each technique alone rarely describes the target fully, it usually encounters problems such as illumination variability and occlusion, which results in appearance variations. So in order to solve these problems, it can be expected that the combination of a few features together, can improve the performance of detection [5].

Color is one of the most important local based features used in machine vision algorithms [9]. It provides useful visionary cues in order to distinguish leaves, branches and

Dor Oppenheim, Yael Edan, and Guy Shani are with the Ben-Gurion University of the Negev, P.O.B. 653 Beer-Sheva 8410501, Israel (e-mail: doropp@post.bgu.ac.il, yael@bgu.ac.il, shanigu@bgu.ac.il).

other objects from the target – usually fruit and vegetable. In a research aimed to estimate flowering in an apple orchard, the white color of the apple flowers was the main cue. The researchers acquired the images at night using artificial lighting so lighting conditions were stable and good for the detection. The researchers used the HSV color space in order to segment the flowers from the background [10]. When acquiring images at day, lighting conditions become a problem. In a study on estimating the number of mango fruit on trees, RGB, NDI and YCbCr color spaces were used in order to perform the segmentation, followed by a blob detector for detecting the round shape of the mango fruit. The authors pointed out that in order to improve detection, emphasis should be on the round shape of the fruit instead of color cues due to illumination variation [11]. When color is not a prominent feature, shape features becomes the main cue for detection e.g., detecting green apple fruit [12] and immature green citrus [13]. The green color of the fruit blends with its foliage and makes it hard to detect. Therefore, circle detection was used in order to detect the round shape of the fruit.

The focus of this research was to develop a real-time, computationally simple tomato flower detection algorithm which will be possible to implement on a drone's simple processor. Therefore, the development of the algorithm was forced to remain simple and fast. Such as in the research of detecting and estimating yield of the yellow lesquerella, where researchers mainly used the HSV color space for the segmentation of flower from background, followed by some simple morphological operation which used simple shape features. The optimal hue values used for the segmentation were 0.12-0.18 as upper and lower bound for the yellow color [14]. Similarly, another research on estimating the number of tangerine white flowers used white color pixels from the HSV color space to perform the segmentation [15]. Both emphasized the problematic variance in natural lighting conditions.

The overall objective of the study was to develop a robust algorithm to correctly detect yellow tomato flowers in variant greenhouse conditions. Specific objectives were to:

1) Develop an adaptive thresholding technique which will be able to deal with variable illumination conditions.
2) Use color information from the image in order to perform segmentation according to lighting conditions.
3) Detect flowers in given images with low false positives and high accuracy.

II. Materials & Methods

A. Image Acquisition

To develop an algorithm for tomato flower detection, 1,350 images were acquired in two greenhouses. The photos were captured using a smartphone's LG-G4 camera and a Canon PowerShot 590IS. The images were captured in the RGB color space, with a resolution of 5312×2988 and 3264×1832 respectively. The first collection was taken in a school greenhouse located in Kibbutz Shoval in south Israel on the 5th of November 2015 and contained 250 images. The second collection, which contained the remaining 1100 images, was taken in a research and development greenhouse owned by Hazera Seeds Ltd. Both image collections included various cultivars of tomato flowers. Since it was important to base developments on real world conditions regardless to tomato flower cultivar, and since the acquisition device for the drone has not been chosen yet, photos of different types of tomato flowers were taken from 3 different angles, at 3 different periods of the day from random distances and at different heights that mimic the drone's location in the greenhouse. The images were acquired from 3 different angles to indicate the optimal location in which to place the camera on the drone. A top view simulates a camera positioned on the bottom side of a drone, a view up front simulates a camera positioned on one of the sides of the drone and a view tilted to the side simulates a camera positioned in the front of the drone (see Fig. 1). The acquisition time was divided into three periods – morning, noon and afternoon in order to create variability in lighting conditions.

Fig. 1 Examples of view points

B. Database

Analysis was conducted for a total of 1069 images that were chosen after filtering bad images. The database consists of approximately 25,000 single tomato flowers. Flowers were tagged manually using MATLAB'S 2014b training image labeler application and the image itself was categorized by its camera type, acquisition angle and acquisition time. Every visible flower in the image was labeled by a rectangle bounding box (Fig. 2). Flowers on plant rows behind the main plant row in the image are usually small and blurry. These flowers were ignored and were not tagged (Fig. 2).

Fig. 2 Examples of tagged images

C. Algorithm

The computer vision algorithm was developed with MATLAB 2014b using its image processing and computer vision toolboxes. Its main procedure is depicted in Fig. 4. First, lighting conditions are calculated and the RGB image is transformed to the HSV color space. Second, the image is segmented into foreground and background using color cues according to lighting conditions, and third, a simple classification is performed on the segmented foreground as to what is flower and what is not according to size and location in the image. The algorithm inputs an RGB image and outputs a list of detected flowers, each described by a connected component and its X and Y location in the image, displayed as a binary image. In the development of each part of the algorithm feasibility of real time was taken into consideration.

1) HSV Transformation

The HSV color space has proven to be useful in many color based algorithms for detection [16]-[19]. In addition, hue (the H component) is an attribute of the pure color of the image scene which is important for the algorithm's color based segmentation and is relatively invariant to lighting conditions. To convert between RGB to HSV color space, the MATLAB command rgb2hsv was used. First, the function normalizes RGB pixel values to range [0,1] by dividing by the bit depth of each channel. Then, the normalized RGB is converted to HSV in the range [0,1].

2) Lighting Condition Estimation

To overcome the various illumination conditions in the greenhouse, illumination conditions are calculated to set accordingly adapted segmentation parameters. The HSV image is the input to this part of the algorithm, and the calculation of the lighting condition is done for the whole image considering all of the pixels in the image. Comparing the HSV histograms of the images (can be seen in Fig. 3), we found that the S component, saturation, distinguished easily between the darker and brighter images. Two indicators were chosen to distinguish between the images – the median value of the saturation and the skewness of the histogram of saturation values.

3) Color Image Segmentation

Since tomato flower's yellow is very distinguishable it was chosen as the main feature to make the first segmentation of the image. The segmentation was performed over the HSV image and considered the lighting conditions calculated before. The hue pixel values were used to distinguish yellow parts of the image from other colors. Saturation pixel values were used to segment very bright parts out of the image because the S component provides useful information on the amount of light returned from the object in the image [9].

Choosing the threshold values in a segmentation process is a crucial part, because each pixel segmented out of the image and considered as background is not taken into consideration in the following steps, even if it is a flower's pixel and vice-versa. So in order to choose the segmentation values over the H and S components correctly, a few steps were performed.

The first step's goal was to choose relevant threshold values for hue segmentation. Two threshold values were derived from a different database of 200 pictures previously taken in a greenhouse located in Berurim, Israel. A sampling program was written using MATLAB 2014b. Ten samples of randomly chosen yellow flower parts in each one of the 200 images were taken, a total of 2,000 samples. The thresholds were then chosen empirically. The low threshold of the hue value chosen was 0.12 and the high threshold was chosen to be 0.18, since it comprises more than 90% of the samples. After some trial and error procedures, it was concluded that these thresholds segment yellow parts relatively well with low noise. Adaptation for the lighting condition was performed after inspecting hue values of flower pixels in both darker images and brighter ones. Darker images had lower hue values than brighter images; therefore, the threshold values of brighter images were set to 0.12-0.18 and for darker image to 0.11-0.17.

The second step was choosing the saturation threshold. High saturation sometimes causes not yellow parts to appear as yellow, so in order to minimize these cases the threshold was set to 0.2 in darker images and 0.4 in brighter images, any S value lower than 0.2 or 0.4 accordingly (highly lighted) was segmented out of the image.

Lastly, the two segmented images of the H and S components are joined together using an AND operator so only the yellow pixels with low saturation continue to the morphological operations. The result of the merge is a binary image, white being the pixels of interest and black being the background.

(a)

(b)

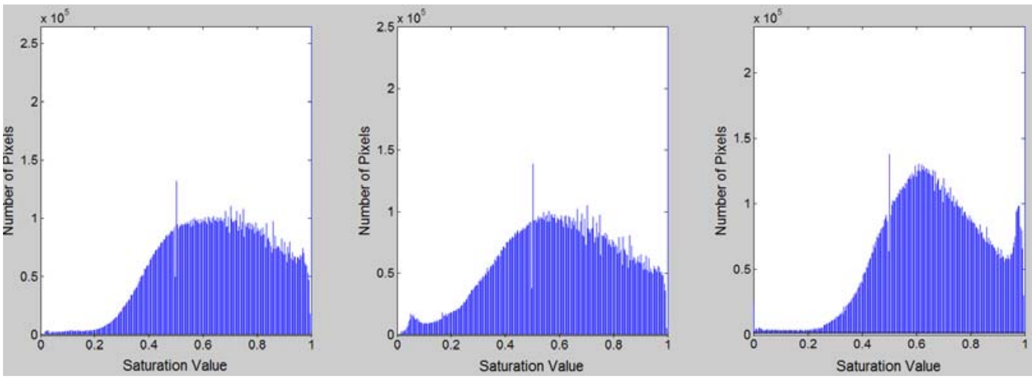
Fig. 3 Examples of bright images (a) and dark images (b)

4) Morphological Operations

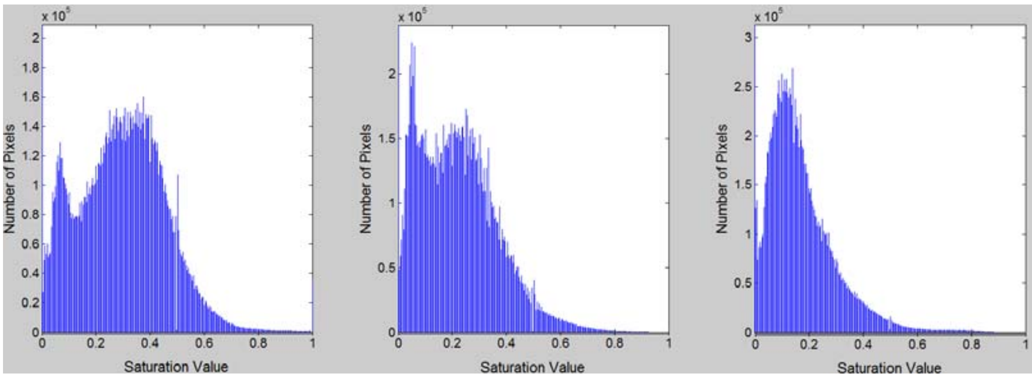
Morphological operations are a collection of non-linear operations related to the geometric shape that neighboring pixels form [20]. In this algorithm, opening and closing operations are carried out, after segmentation is done. Opening removes small objects in the foreground, whereas closing removes small "holes" in the foreground. Segmentation usually leaves small patches of noise in the image and holes in the foreground caused by the variations in lighting conditions and shading. In order to remove these noises, first an opening procedure is done so as to remove small noises, followed by a closing procedure to remove "holes".

5) Classification

The final step of the algorithm is a size feature classification in which it eliminates small connected components. This elimination removes small objects that have a very small probability to be a flower or a faraway flower that we do not seek yet. The classification is done in two steps. The first one extracts the connected components from the binary image using MATLAB's bwconncomp function, which returns the connected component as a vector of objects. And the second one removes any object according to its area. The area of an object is considered as the number of pixels that it consists of. Although the image was segmented and morphological operations have been done, still there is usually large amount of objects that are actually noise. In order to deal with them, the final elimination is done according to the amount of connected components in the image. The connected components left in the image are considered as yellow tomato flowers.



(a)



(b)

Fig. 4 Saturation histograms of bright images (a) compared to darker images (b) from Fig. 3

III. Results

The presented algorithm in this study was tested with the 1069 validation images with various lighting conditions, occlusions and different acquisition angle and times. Fig. 6 shows an example of the algorithm's main steps results. Fig. 6 (a) shows an example of an original image. Fig. 6 (b) shows a form of display of the HSV color map. The similarity of the yellow and green parts is salient in this image. The reason is that the hue values of the color green and the color yellow are very similar, which adds complexity to the segmentation task. Fig. 6 (c) shows the output of the color segmentation step. Many pixels (white parts in the image) passed through the segmentation process even though they do not appear as completely yellow in the RGB image. Fig. 6 (d) displays the result of the noise removal. Fig. 6 (e) shows the final result after classification has been done.

The algorithm was tested on 1069 images. Analyses of the algorithm's results were performed in RStudio. The goal of the analysis was to evaluate the performance of the algorithm according to the angle of acquisition and time of acquisition and global performance disregarding acquisition angle and time. The Precision-Recall curve was used for visualization and analyses of the results. The precision-recall curve is created by plotting the results of the mean precision and recall of each threshold value that was used to segment the yellow flower pixels. High values of precision are received when algorithm's thresholds are close to one another, and low values are received when the thresholds are farther apart. Recall reacts opposite to precision when changing threshold values.

A. Angle of Acquisition Analysis

Determining the angle of acquisition is an important part of developing the drone pollinator. The results of this analysis can determine where the camera will be attached to the drone. As can be seen in Fig. 6, the best view point angle was the front view. These results indicate that attaching the camera on the side of the drone, when facing the plants would help generate better results than other angles. The results are as expected since images facing the plants mostly include the plant and flowers solely, whereas images from other angles include part of the greenhouse and paths in the greenhouse which makes the segmentation process more complex.

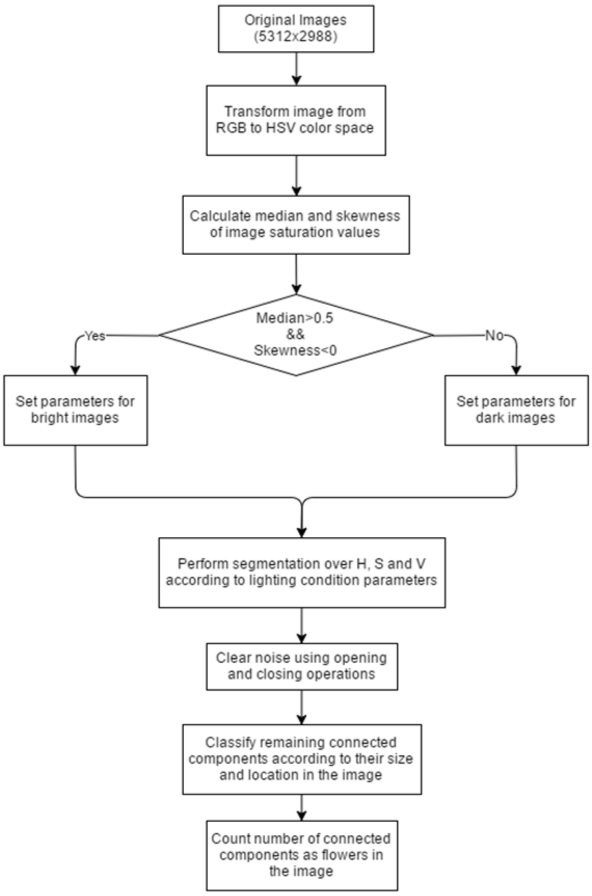


Fig. 5 Tomato flower detection algorithm flow chart

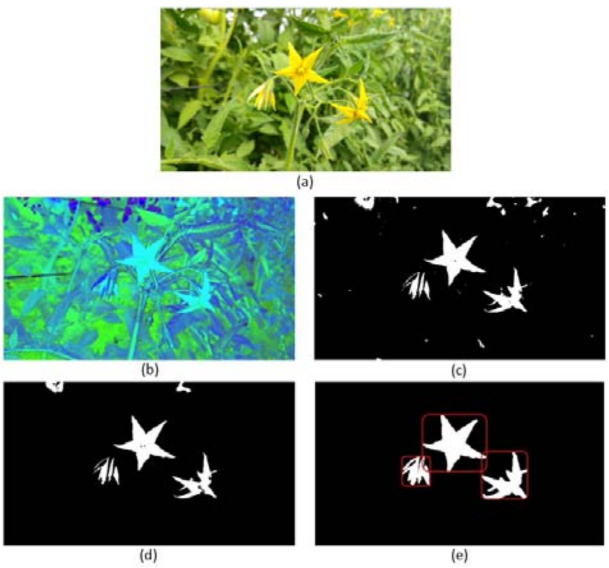


Fig. 6 An example of the main procedure of the proposed algorithm: (a) Original image, (b) HSV image, (c) After segmentation over H and S, (d) After noise removal, (e) Classification results (red rectangles are connected components classified as flowers)

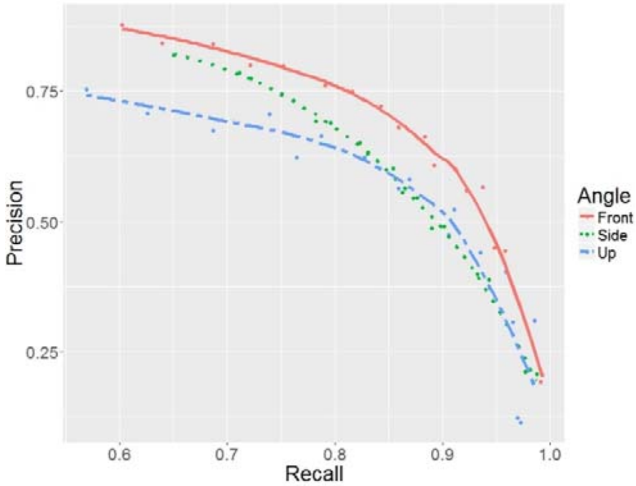


Fig. 7 Precision-Recall Curve by Acquisition Angle

B. Time of Acquisition Analysis

Acquisition time was divided into three categories, in accordance with the three periods of the day (morning, noon and afternoon), in which each image was acquired at. Using a fully adaptive algorithm should not show any differences between acquisition times. However, Fig. 7 shows that acquiring images in the afternoon had mostly better results than other periods of the day. Fig. 7 also shows that as the day proceeds detection of the tomato flowers improves. A possible explanation is that the fixed threshold values for detecting yellow parts in the image are better fitted to lighting conditions later in the day. Including time as a parameter in the algorithm could improve detection.

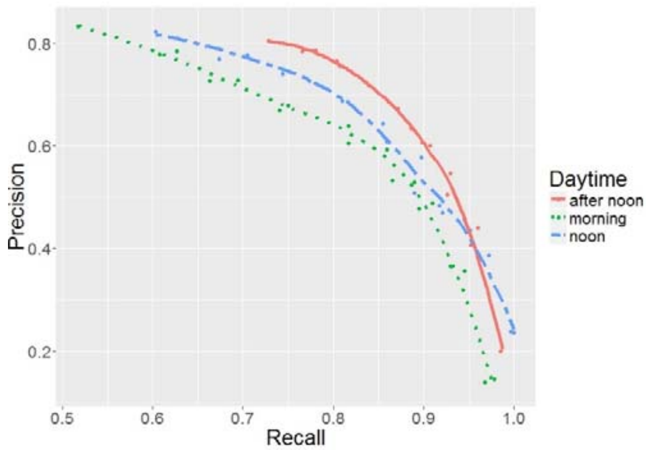


Fig. 8 Precision-Recall Curve by Acquisition Time

IV. Conclusion and Future Work

Images taken in the afternoon from an angle facing the plants provided better results in precision and recall than any other angle. Optimal hue values for detecting the yellow flowers were found as well. Further analysis of the results shows that when using the best performing parameters from this study, that is to say, front acquisition angle, optimal hue threshold values and afternoon acquisition times. Precision and recall results increased from 74% and 75% respectively to 80% for both performance indicators. This type of tomato flower detection was not tested before. Ongoing research is aimed to improve the detection algorithm using the large database created for this research by implementing machine learning algorithms and a local adaptive threshold for better segmentation and detection rate. Future research will include images acquired from the drone pollinator for mimicking real conditions of pollination tasks. In addition, a flower counting feature will be tested in order to provide continuous monitoring for the drone.

REFERENCES

[1] K. Kapach, E. Barnea, R. Mairon, Y. Edan, O. Ben-Shahar, and O. Ben Shahar, "Computer vision for fruit harvesting robots state of the art and challenges ahead," *Int. J. Comput. Vis. Robot.*, vol. 3, no. 1/2, p. 4, 2012.
[2] S. A. Cameron, J. D. Lozier, J. P. Strange, J. B. Koch, N. Cordes, L. F. Solter, and T. L. Griswold, "Patterns of widespread decline in North American bumble bees," *Proc. Natl. Acad. Sci.*, vol. 108, no. 2, pp. 662–667, 2011.
[3] D. a Sumner and H. Boriss, "Bee-conomics and the Leap in Pollination Fees the Supply and Demand Issues and Operation of the Pollination Market," pp. 9–11, 2006.
[4] R. Ward, A. Whyte, and R. R. James, "A Tale of Two Bees: Looking at Pollination Fees for Almonds and Sweet Cherries," *Am. Entomol.*, vol. 56, no. 3, pp. 170–177, 2010.
[5] R. Wood, R. Nagpal, and G.-Y. Wei, "Flight of the Robobees," *Sci. Am.*, vol. 308, no. 3, pp. 60–65, 2013.
[6] A. Gongal, S. Amatya, M. Karkee, Q. Zhang, and K. Lewis, "Sensors and systems for fruit detection and localization: A review," *Comput. Electron. Agric.*, vol. 116, pp. 8–19, 2015.
[7] A. R. Jiménez, R. Ceres, and J. L. Pons, "a Survey of Computer Vision Methods for Locating Fruit on Trees," *Trans. ASAE*, vol. 43, no. 6, pp. 1911–1920, 2000.
[8] D. R. Martin, C. C. Fowlkes, and J. Malik, "Learning to detect natural image boundaries using local brightness, color, and texture cues," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 5, pp. 530–549, 2004.
[9] H. D. Cheng, X. H. Jiang, Y. Sun, and J. Wang, "Color image segmentation: Advances and prospects," *Pattern Recognit.*, vol. 34, no. 12, pp. 2259–2281, 2001.
[10] M. Hočevar, B. Širok, T. Godeša, and M. Stopar, "Flowering estimation in apple orchards by image analysis," *Precis. Agric.*, vol. 15, no. 4, pp. 466–478, 2014.
[11] A. Payne, K. Walsh, P. Subedi, and D. Jarvis, "Estimating mango crop yield using image analysis using fruit at 'stone hardening' stage and night time imaging," *Comput. Electron. Agric.*, vol. 100, pp. 160–167, 2014.
[12] R. Linker, O. Cohen, and A. Naor, "Determination of the number of green apples in RGB images recorded in orchards," *Comput. Electron. Agric.*, vol. 81, pp. 45–57, 2012.
[13] C. Zhao, W. S. Lee, and D. He, "Immature green citrus detection based on colour feature and sum of absolute transformed difference (SATD) using colour images in the citrus grove," *Comput. Electron. Agric.*, vol. 124, pp. 243–253, 2016.
[14] K. R. Thorp and D. a. Dierig, "Color image segmentation approach to monitor flowering in lesquerella," *Ind. Crops Prod.*, vol. 34, no. 1, pp. 1150–1159, 2011.
[15] U. Dorj, K. Lee, and M. Lee, "A computer vision algorithm for tangerine yield estimation," *Int. J. Bio-Science Bio-Technology*, vol. 5, no. 5, pp. 101–110, 2013.
[16] A. Aquino, B. Millan, S. Gutiérrez, and J. Tardáguila, "Grapevine flower estimation by applying artificial vision techniques on images with uncontrolled scene and multi-model analysis," *Comput. Electron. Agric.*, vol. 119, pp. 92–104, 2015.
[17] N. Bairwa and N. K. Agrawal, "Counting of Flowers using Image Processing," *Int. J. Eng. Res. Technol.*, vol. 3, no. 9, pp. 775–779, 2014.
[18] R. Kaur and S. Porwal, "An Optimized Computer Vision Approach to Precise Well-Bloomed Flower Yielding Prediction using Image Segmentation," *Int. J. Comput. Appl.*, vol. 119, no. 23, pp. 15–20, 2015.
[19] A. Plebe and G. Grasso, "Localization of spherical fruits for robotic harvesting," *Mach. Vis. Appl.*, vol. 13, pp. 70–79, 2001.
[20] L. Lucchese and S. K. Mitray, "Color image segmentation: A state-of-the-art survey," *Proc. Indian Natl. Sci. Acad. (INSA-A). Delhi, Indian Natl Sci Acad*, vol. 67, pp. 207–221, 2001.