Available online at www.sciencedirect.com

ScienceDirect

IFAC PapersOnLine 51-17 (2018) 185–190

Kiwifruit yield estimation using image processing by an Android mobile phone

Longsheng Fu*. Zhihao Liu. Yaqoob Majeed***. Yongjie Cui****.**

* College of Mechanical and Electronic Engineering, Northwest A&F University, Yangling 712100, China
Key Laboratory of Agricultural Internet of Things, Ministry of Agriculture, Yangling 712100, China
Shaanxi Key Laboratory of Agricultural Information Perception and Intelligent Service, Yangling, 712100, China
Center for Precision&Automated Agricultural Systems, Washington State University, Prosser, WA 99350, USA
(**Corresponding Author**, Tel: +86-2987092391; e-mail: fulsh@nwafu.edu.cn, longsheng.fu@wsu.edu)
** College of Mechanical and Electronic Engineering, Northwest A&F University, Yangling 712100, China
(e-mail: lzhdelightsome@foxmail.com)
*** Center for Precision and Automated Agricultural Systems, Washington State University, Prosser, 99350, USA
(e-mail: yaqoob.majeed@wsu.edu)
**** College of Mechanical and Electronic Engineering, Northwest A&F University, Yangling 712100, China
(e-mail: cuiyongjie@nwafu.edu.cn)

Abstract: Advanced electronic technology makes mobile phone suitable for image processing technology on fruit yield estimation. A new method was proposed to estimate yield of kiwifruit orchard via image processing using an Android mobile phone. Image processing was used to select significant color channels and their threshold for segmentation, generate binarized image, and remove noise. A fruit-counting algorithm was developed to calculate the number of kiwifruit by identifying the number of fruit calyxes. Image area was estimated by the pixels proportion of a reference board hung under the canopy. Finally, yield was calculated by multiplying the proportion of the image in three-dimensional space with the number of kiwifruit in an image and the kiwifruit orchard area. Under natural conditions, the performance of the fruit counting algorithm software was validated with 100 kiwifruit images. The proposed method achieved a good precision and recognition ratio of 76.4%. Results showed the pixels area of the reference board could be correctly detected. It is reliable to estimate the yield of kiwifruit orchard by recognizing the number of kiwifruit and the area of the reference board. The results indicated that the approach could be utilized to estimate the fruit yield of a kiwifruit orchards, which is valuable information to forecast yield, and plan harvest schedules.

Keywords: kiwifruit; yield estimation; image processing; Android mobile phone; fruit calyx

1 INTRODUCTION

Production forecasting is an important factor in precision management of orchard (Gong et al., 2013). It provides the guidance to farmers on product price and effectively managing the market variables. With the decrease in cost of hardware devices and the rapid progress in the field of artificial intelligence (Panian, 2009), fruit yield estimation devices are becoming smaller and simpler to use (Cubero et al., 2018).

Yield estimation of fruit would provide valuable information for generating prescription maps for precision management (Bargoti & Underwood, 2017). During the past few decades, there had been several studies on fruit yield estimation. Wang et al. (2013) used an autonomous orchard vehicle with two-camera rig to take pictures and estimate apple orchard yield by developing new apple detection algorithms on computer. Dorj et al. (2013) applied digital camera to capture images, remove noises divide images into sub-images and detect the white pixels to estimate the yield of Tangerine flower. Schumann et al. (2006) estimated citrus fruit yield by using ultrasonic sensors to measure the size of citrus trees. Ye et al (2006) carried out a preliminary step to develop a neural network model for the estimation of tree yield from airborne hyperspectral image. However, these methods require expensive and cumbersome measuring equipment, which is inconvenient for small production areas. Especially in China where most orchards are in small scale, which requires low cost measuring equipment.

The research on the fruit yield estimation is not only important for individual farmers (Moonrinta et al., 2011), but also has a profound impact on the development of agricultural management policies at regional and national levels. However, there is currently no study on the estimation of kiwifruit production. Although some researches were conducted on kiwifruit image recognition using ordinary camera to capture fruit images and developing image processing algorithm based on common computer, it mainly aims for fruit robotic harvesting which normally equipped with high performance computers (Fu et al., 2015, 2017, 2018).

It is important to introduce a simple, cheap and convenient method of estimating kiwifruit yield, which can not only accurately estimate yield but also be used for real-time applications. This article has developed an Android mobile

2405-8963 © 2018, IFAC (International Federation of Automatic Control) Hosting by Elsevier Ltd. All rights reserved.
Peer review under responsibility of International Federation of Automatic Control.
10.1016/j.ifacol.2018.08.137

phone (AMP)-based kiwifruit production estimation system. For this purpose, an image processing-based fast fruit counting algorithm was developed for AMP.

2 IMAGE ACQUISITION

Kiwifruits are commercially grown on T-bar sturdy support structures (Hopping et al., 1993). The T-bar trellis is common in China because of its low cost. It consists of a 1.7 m high post and an approximately 1.7 m wide cross arm (Fu et al., 2017). RGB images of the fruits were captured by placing the AMP on a selfie stick with the rear lens under the canopy. A reference board that covered by a known-size coordinate paper was hung under the canopy for calculating space area. All the images were captured in late October during the harvesting season on the most common cultivar ‘Hayward’ at Meixian Kiwifruit Experimental Station (34°07'39"N, 107°59'50"E, and 648 m in altitude) at the Northwest A&F University. Four AMPs, model Millet 5, Millet 4, and Millet 4c from Xiaomi Science and Technology Co. Ltd. (China) and HuaWei Honor 6 plus from HuaWei Technologies Co. Ltd. (China) were used for this study. These module incorporate Qualcomm snapdragon series CPU and Kirin series, as shown in Table 1.

Table 1 Four different types of AMP

Mobile phone types	Millet 5	Millet 4c	Millet 4	HuaWei Honor 6 plus
CPU and main frequency	Qualcomm snapdragon 820 2.15 GHz	Qualcomm snapdragon 808 1.8 GHz	Qualcomm snapdragon 801 2.5 GHz	Kirin 925 1.8 GHz

They are operated by Android operating system and a megapixel video camera associate with LED flashlight are included. They can capture the images with high resolution. The four AMPs were placed under the shelf of around 1 m to take images of kiwifruit with Auto-Exposure mode. The acquired image were shown in Fig. 1. A total of 60 images were taken in the morning (20 images with flash on and 40 images with flash off) and 40 images in the afternoon (20 images with flash on and 20 with flash off).

(a) Morning without flash (b) Morning with flash (c) Afternoon without flash (d) Afternoon with flash

Fig. 1 Acquired images by the AMP in the morning and afternoon

3 SYSTEM DEVELOPMENT

The kiwifruit yield estimation system includes image capturing, image processing, fruit counting, marker recognition, and yield estimation components. The algorithms have been implemented for AMP using the open-source BSD-licensed library OpenCV (https://en.wikipedia.org/wiki/BSD_licenses), the open software development kit (SDK) for Android (http://developer.android.com/sdk/terms.html), and the programming environment Eclipse (http://www.eclipse.org/org/documents/epl-v10.php) using Java language to complete the algorithms and man-machine interface.

3.1 Image preprocessing

Pixels of fruits were sampled to study fruit color for segmentation using the MATLAB 2016a (Mathworks, 2016). Ten pixels of each fruit were selected by randomly clicking on the fruit area and their color information (Red, Green, and Blue) were saved in a data file. In total, 5790 pixels of fruits were collected. Then these RGB values were converted to HSV (Hue, Saturation, and Value) suing the MATLAB function ‘rgb2hsv’. The advantage of HSV is that each of its attributes corresponds directly to basic color concepts, which makes it conceptually simple (Skoneczny, 2012). And the L*a*b* color spaces were also calculated.

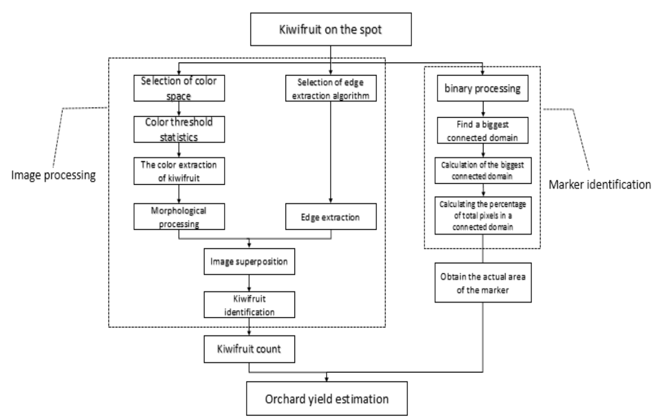


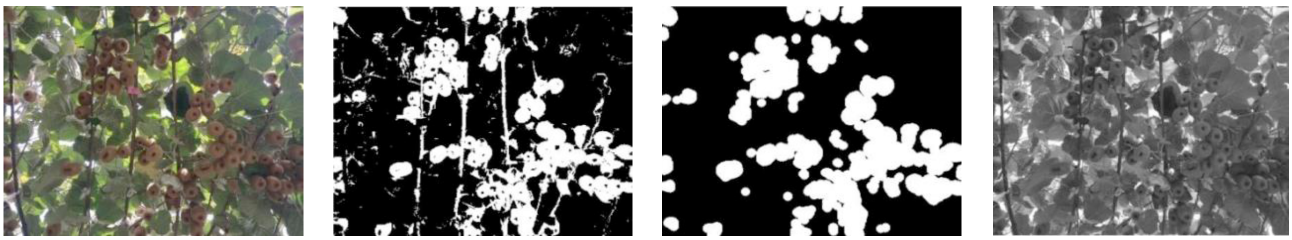
Fig. 2 Kiwifruit yield estimation software system development

Distributions of the 5,790 sampled kiwifruit pixels under different color channels were analyzed and normalized for comparison. The color information of kiwifruit pixels is analyzed in RGB, HSV and L*a*b* color space, and the distribution characteristics of the different channels of these color spaces were evaluated by the kurtosis value to select the significant color spaces and ranges for fruit segmentation using the equation 1. Where N is the number of kiwifruit pixels, Y_i is the corresponding value of the single pixel i in channels, Y' is the average value of pixels in the same channel, S is the standard deviation of pixels in the same channel.

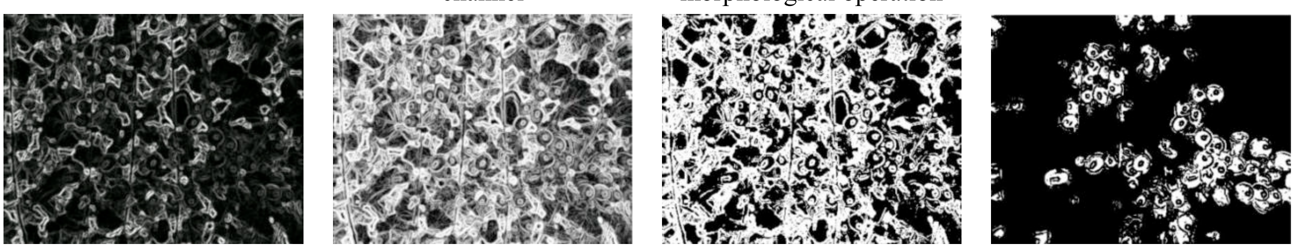
$$K = \frac{\sum_{i=1}^{N}(Y_i - Y'\ )^4}{(N-1)S^4} \tag{1}$$

With selected color spaces and ranges, the image was segmented by subtracting channels in the corresponding color space from each pixel, and selecting those pixels which were within a given threshold, as shown in Fig. 3b, where pixels in the threshold range are white (255), otherwise black (0). In order to remove noises, the image was processed by morphological operation corrosion with a circle of radius 3 then dilation with a circle of radius 6, as shown in Fig. 3c. On the other hand, the original RGB image (Fig. 3a) was converted to gray image, as shown in Fig. 3d using Equation (2). To obtain the edges of kiwifruit, Canny operator was used (Ding & Goshtasby, 2001). And then, Fig. 3e was equalized using histogram equalization (Pizer et al., 1987) to improve contrast, as shown in Fig. 3f. The Otsu threshold method (Xu et al., 2011) was used to convert Fig. 3f to a binary image, as shown in Fig. 3g. Finally, AND operation was applied on Fig. 3c and Fig. 3g bit by bit to speed up the image processing algorithm. Then the fruit area with the central fruit calyx was obtained, as shown in Fig. 3h.

$$\text{Gray} = R\times0.299 + G\times0.587 + B\times0.114 \tag{2}$$



(a) Original kiwifruits image (b) Segmented image by threshold in H and a* channel (c) Fruits area image by eliminating noises using a morphological operation (d) Gray kiwifruits image



(e) Fruits boundary obtained by Canny operator (f) Histogram equalization (g) Segmented image by Otsu threshold (h) Implement bit AND operation on (c) and (g)

Fig. 3 The flow of kiwifruit image preprocessing

3.2 Kiwifruit calyx detecting and counting

After image preprocessing, fruits were counted by recognizing the number of fruit calyxes. A circle template was scanned over the image at the segmentation and edge detection date. To increase the speed, it was only performed within the segmented areas. The template analyze the predetermined radial points around the circle's perimeter, as shown in Fig. 4. This template encompasses the fruit's calyx and initially only the periphery points were assessed.

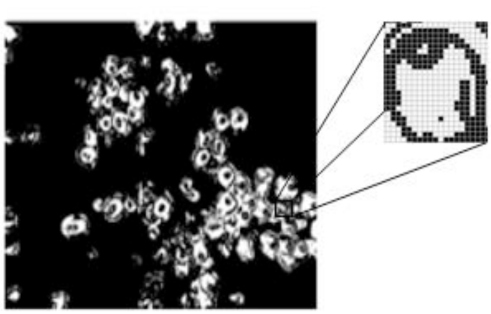


Fig. 4 The white pixels that are scanned

When the template was positioned over a fruit in the image, the fruits blossom end edge points were the majority of points forming a pattern within the template. Final template point validation was determined from the uniformity of edge data variance, using a width/height ratio which was used to assess dispersal uniformity. As the width/height ratio approaches to 1, the likelihood of desired circular point dispersal increased. Meeting the weight/height point dispersal criteria, the point fruit was recognized. Then the fruit number was counted as the number of recognized fruit calyx.

3.3 Reference board area detection

The Sobel edge detection (Lang & Zheng, 2010) function in OpenCV was used to detect the obvious edge of the rectangular area of the known-size reference board. And then the contour detection is carried out. The size of the reference object is hold by calculating the size and shape complexity of each contour.

(a) Kiwifruit image with reference board (b) The detected reference outline image

Fig. 5 Original image and processed image with reference board

3.4 Fruit density estimation

The fruit density was estimated using the Equation (3). Kiwifruit image total pixel number S and reference board pixel P are used to get the proportion of reference material in kiwifruit image. Then the known-size Q of the reference board is used to divide the results to get the actual area of images in three-dimensional space. Finally the number of detected kiwifruit number M is used to get the approximate density L of fruit distribution in the orchard. Finally, the fruit yield can be obtained by multiplying the fruit density with orchard area.

$$L = [(P \div S) \div Q] \times M \qquad (3)$$

4 RESULTS AND DISCUSSION

4.1 Selection of color channels and determination of threshold

The kurtosis values were calculated for distribution characteristics of different color channels in the RGB, HSV, L*a*b* color spaces, respectively. The pixel and color information of kiwifruit under 5,790 different space-time conditions were normalized and the color distribution were shown in Fig. 6. The distribution of color in few color channels is more aggregated as compared to other channels.

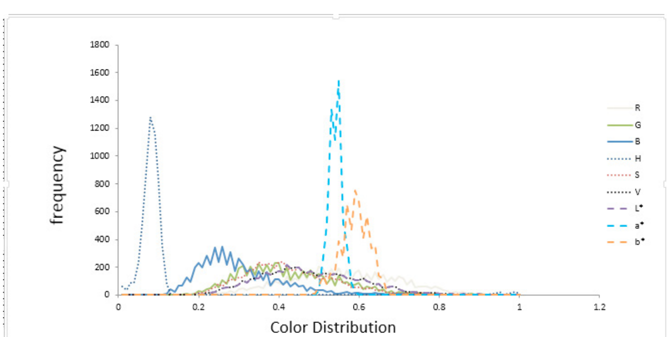


Fig. 6 Distribution of kiwifruit image in different color channels

The quantitative description of the color distribution based on the kurtosis value was shown in Table 2. The H channel in HSV space and a* channel in L*a*b* had heavy tails. Their kurtosis values were 19.5 and 21.3 respectively, which were higher than those of other channels. Therefore, the H and a* channel were selected for fruit segmentation and their ranges were 0 to 48 and 0 to 22 respectively.

4.2 performance of the fruit calyx recognition algorithm

The number of fruits counted by the fruit calyx recognition algorithm was compared with the actual number of fruits in the region covered in the image. The algorithm was tested in different kiwifruit images, and the data in Table 3 show that the errors between the numbers of fruits counted by the fruit calyx recognition algorithm and counted by human observers were small with an average estimation ratio of 76.4%. The main reasons for this error are as follows:

1) Kiwifruit calyx may be blocked by leaves and branches and cannot be correctly identified.

2) Some kiwifruit calyx are oval, resulting in the width/height ratio is too big or too small when the circle template was positioned over the fruits in the image. This makes the algorithm cannot detect the actual kiwifruits.

Table 2 Kurtosis values of the distribution of kiwifruit images in different color channels

Color space	RGB			HSV			L*a*b*		
Channel	R	G	B	H	S	V	L*	a*	B*
Kurtosis value	-1.1	-0.7	1.4	19.5	8.0	7.5	3.7	21.3	6.3

4.3 Performance of the fruit calyx recognition algorithm

The number of fruits counted by the fruit calyx recognition algorithm was compared with the actual number of fruits in the region covered in the image. The algorithm was tested in different kiwifruit images, and the data in Table 3 show that the errors between the numbers of fruits counted by the fruit calyx recognition algorithm and counted by human observers were small with an average estimation ratio of 76.4%. The main reasons for this error are as follows:

3) Kiwifruit calyx may be blocked by leaves and branches and cannot be correctly identified.

4) Some kiwifruit calyx are oval, resulting in the width/height ratio is too big or too small when the circle template was positioned over the fruits in the image. This makes the algorithm cannot detect the actual kiwifruits.

4.4 Fruit calyx recognition in different conditions

In order to test the algorithm in different natural conditions, 100 images were used to test the algorithm. The results of kiwifruit calyx recognition were divided into 3 categories: true recognized, false recognized, and unrecognized, as shown in Table 3. The acquired images with the Auto-Exposure mode in the morning showed an exposure time of 1/50 s and ISO of 64, while that on the afternoon showed 1/33 s and 250, respectively. It meant that the AMP camera system required more light in the afternoon than morning to capture an image, which caused brighter images in the afternoon, as shown in Fig. 1. The fruit images in the afternoon were clearer than that in the morning and thus achieve higher true recognized results 77.9% of afternoon than that (74.2%) of morning. The false recognized and unrecognized results were lower in the afternoon than that in the morning. For different conditions, the images acquired with flash has obtained higher true recognized results than without flash. This indicates that it is necessary to provide artificial light even in the daytime, which could help to increase the color contrast between fruit calyx and skin, and background, as shown in Fig. 1.

Table 3 Recognition results of the kiwifruit images with AMP

Time	No. of images	Flash	Exposure time (s)	ISO	True recognized	False recognized	Unrecognized
Morning	40	No	1/50	64	73.9%	12.0%	26.1%
	20	Yes	1/50	64	74.6%	11.4%	25.4%
Sub total	60		1/50	64	74.2%	11.8%	25.8%
Afternoon	20	No	1/33	250	77.5%	9.2%	22.5%
	20	Yes	1/33	250	78.3%	8.5%	21.7%
Sub total	40		1/33	250	77.9%	8.8%	22.2%
Total	100				76.4%	10.3%	23.6%

4.5 Recognition of reference board

In order to test the reference board recognition accuracy, 20 images with reference board are randomly chosen to test under different conditions. To get the actual area of reference board, the reference board in 20 images is cut accurately along the board, using Photoshop (support accurate positioning). Based on the relative error formula, the error between the actual area of the reference board and the area of the reference board obtained by the algorithm is analyzed. The results of error rate are low than 1.03%, the average is 0.76%. This indicated that the result of the reference board area algorithm is accurate and satisfy the actual kiwifruit yield estimation need.

4.6 Fruit yield estimation

The estimated kiwifruit yield was calculated by multiplying kiwifruit orchard area with the kiwifruit density. According to the regular outline and obvious edges, the kiwifruit density is given by selecting the largest connected area and the most regular outline. The average of the estimated yield in October 2016 is 80 kiwifruits per square meter. Compared with the actual production of the kiwifruit orchard in the same year, the results of kiwifruit estimation yield is a little lower than the actual yield 106 kiwifruits per square meter. The reason is that the kiwifruit orchards are unevenly planted and the actual kiwifruit orchard area may be smaller than the record. In addition, during the picking process, people accidentally knocked the kiwifruit off. This is also a factor in the inaccurate estimation of kiwifruit production. Overall, the results of kiwifruit production estimates are generally accurate. It can be used to estimate the output of kiwifruit for the farmer.

4.7 Running time

The whole algorithm was implemented in Android Studio and ran on AMPs. Processing time is a major concern in a real-time machine vision application. To value the speed of the algorithm on AMPs, 10 randomly chosen images are tested in 4 different types AMPs, as shown in Table 1. Frequency of CPU is a main factor affecting the processing speed of the image algorithm. Millet 5, with the best performance of CPU (2.15 GHz), spend about 2.0 s in average to process each image. The speed of the algorithm is inversely proportional to the performance of CPU. If the algorithm is tested with a mobile phone with a better CPU, the time will be further shortened. In addition, the optimization of the AMP system performance slightly affects the image processing speed. Before the program runs, the algorithm is faster by eliminating the system's redundancy and deleting background programs.

In general, the performance of hardware and software in AMP has been developing rapidly, which future AMP will have cheaper prices and better hardware and more optimized systems, thus, will further improve the image processing speed.

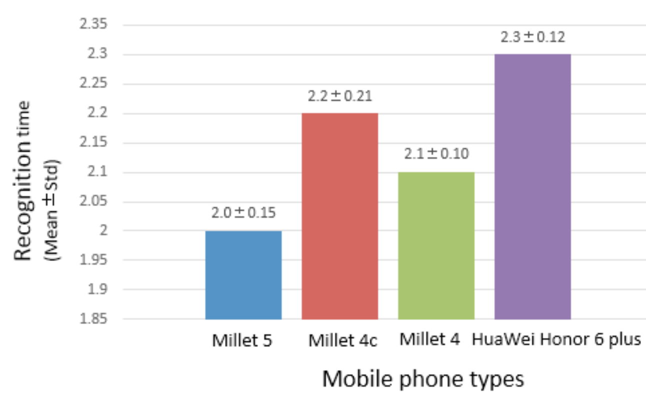


Fig. 7 Running time of algorithm in different AMPs

5 CONCLUSIONS

A methodology using AMP for estimating the yield of kiwifruit was introduced in this work. The fruit-counting algorithm based on kiwifruit calyx suggested that it was able to obtain sufficient statistical precision. A reference board was hung under the canopy to assist estimating image area in the three-dimensional space. And then fruit yield was estimated by calculating fruit density with kiwifruit orchard area. The results showed that the fruit counting algorithm contributed a satisfactory performance with 76.4% accuracy rate under natural conditions. Based on the results, it can be concluded that it is possible to estimate the yield of kiwifruit orchard using the AMP. Thus, this study presents a new method based on portable device for future kiwifruit yield prediction. Yield estimates of kiwifruit orchard obtained by the developed methodology would provide valuable information for planning harvest schedules and generating prescription maps for site application of precision management.

ACKNOWLEDGEMENTS

The work was supported by the Key Research and Development Program in Shaanxi Province of China (Project No. 2017NY-164), and the Science and Technology Innovation Project in Shaanxi Province of China (grant number 2015KTCQ02-12), and the International Scientific and Technological Cooperation Foundation of Northwest A&F University (A213021505). The "Young Faculty Study Abroad Program" of Northwest A&F University Scholarship sponsors Dr. Longsheng Fu in conducting post-doctoral research at Centre for Precision and Automated Agricultural Systems (CPAAS), Washington State University (WSU). The University of Agriculture, Faisalabad (UAF) of Pakistan sponsors Yaqoob Majeed in conducting Ph.D. dissertation studies at WSU CPAAS. Any opinions, findings, conclusions, or recommendations expressed in this publication are those of the authors and do not necessarily reflect the views of WSU.

REFERENCES

Bargoti, S., & Underwood, J. P. (2017). Image segmentation for fruit detection and yield estimation in apple orchards. *Journal of Field Robotics, 34*(6), 1039-1060.

Cubero, S., Albert, F., Prats-Moltalbán, J. M., Fernández-Pacheco, D. G., Blasco, J., & Aleixos, N. (2018). Application for the estimation of the standard citrus colour index (CCI) using image processing in mobile devices. *Biosystems Engineering, 167*, 63-74.

Ding, L., & Goshtasby, A. (2001). On the Canny edge detector. *Pattern Recognition, 34*(3), 721-725.

Dorj, U. O., Lee, K. K., & Lee, M. (2013). A computer vision algorithm for Tangerine yield estimation. *International Journal of Bio-Science and Bio-Technology, 5*(5), 101-110.

Fu, L., Feng, Y., Elkamil Tola, Liu, Z., Li, R., & Cui, Y. (2018). Image recognition method of multi-cluster kiwifruit in field based on convolutional neural networks. *Transactions of the Chinese Society of Agricultural Engineering, 34*(2), 205-211.

Fu, L., Sun, S., Vázquezarellano, M., Li, S., Li, R., & Cui, Y. (2017). Kiwifruit recognition method at night based on fruit calyx image. *Transactions of the Chinese Society of Agricultural Engineering, 33*(2), 199-204.

Fu, L., Wang, B., Cui, Y. J., Shuai, S., Gejima, Y., & Kobayashi, T. (2015). Kiwifruit recognition at nighttime using artificial lighting based on machine vision. *International Journal of Agricultural & Biological Engineering, 8*(4), 52-59.

Gong, A., Yu, J., He, Y., & Qiu, Z. (2013). Citrus yield estimation based on images processed by an Android mobile phone. *Biosystems Engineering, 115*(2), 162-170.

Hopping, M. E., Martyn, J. A. K., & Hacking, N. J. A. (1993). Comparison of growth and yield of kiwifruit on different vine support structures. *New Zealand Journal of Experimental Agriculture, 21*(4), 295-301.

Lang, Y., & Zheng, D. (2010). *An improved Sobel Edge Detection operator*. Paper presented at the IEEE International Conference on Computer Science and Information Technology.

Moonrinta, J., Chaivivatrakul, S., Dailey, M. N., & Ekpanyapong, M. (2011). *Fruit detection, tracking, and 3D reconstruction for crop mapping and yield estimation.* Paper presented at the International Conference on Control Automation Robotics & Vision.

Panian, Z. (2009). *Expected progress in the field of business intelligence.* Paper presented at the Wseas International Conference on Artificial Intelligence, Knowledge Engineering and Data Bases.

Pizer, S. M., Amburn, E. P., Austin, J. D., Cromartie, R., Geselowitz, A., Greer, T., Zimmerman, J. B. (1987). Adaptive histogram equalization and its variations. *Computer Vision Graphics & Image Processing, 39*(3), 355-368.

Puder, A., & Antebi, O. (2013). Cross-Compiling Android applications to iOS and Windows Phone 7. *Mobile Networks & Applications, 18*(1), 3-21.

Schumann, A. W., & Zaman, Q. U. (2006). Estimation of citrus fruit yield using ultrasonically-sensed tree size. *Applied Engineering in Agriculture, 22*(1), 39-44.

Singh, B. K., Thoke, A. S., Verma, K., & Chandrakar, A. (2012). Image information retrieval from incomplete queries using color and shape feature. *Signal & Image Processing, 2*(4), 213-220.

Skoneczny, S. (2012). Nonlinear image sharpening in the HSV color space. *Przeglad Elektrotechniczny, 88*(2), 140-144.

Wang, Q., Nuske, S., Bergerman, M., & Singh, S. (2013). Automated crop yield estimation for apple orchards. *88*, 745-758.

Xu, X., Xu, S., Jin, L., & Song, E. (2011). Characteristic analysis of Otsu threshold and its applications. *Pattern Recognition Letters, 32*(7), 956-961.

Ye, X., Sakai, K., Garciano, L. O., Asada, S. I., & Sasao, A. (2006). Estimation of citrus yield from airborne hyperspectral images using a neural network model. *Ecological Modelling, 198*(3), 426-432.

Zhang, Z. L., & Yi, Z. (2006). Letters: Extraction of a source signal whose kurtosis value lies in a specific range. *Neurocomputing, 69*(7), 900-904.