Sensors and Materials, Vol. 35, No. 9 (2023) 3715–3729
MYU Tokyo

S & M 3446

Development of Size Estimation Method for Occluded Circular Masks and Application to Infrared Peach Images

Ryota Nomura,[1*] Kazuo Oki,[1,2] and Daisuke Takata[3]

[1]Institute of Industrial Science, The University of Tokyo, 1-6-4 Komaba, Meguro-ku, Tokyo 153-8505, Japan
[2]Kyoto University of Advanced Science, 18 Gotanda-cho, Yamanouchi, Ukyo-ku, Kyoto 615-8577, Japan,
[3]Faculty of Food and Agricultural Sciences, Fukushima University,
1 Kanayagawa, Fukushima City, Fukushima 960-1296, Japan

(Received October 30, 2022; accepted June 6, 2023; online published July 6, 2023)

Keywords: geometry, occlusion, radius of curvature, size estimation, infrared camera

In-field fruit size monitoring is useful for estimating fruit maturation and size distribution, making harvesting and marketing decisions, estimating profit, and controlling taste. Many algorithms have been proposed for size estimation, but estimation accuracy for occluded fruits (e.g., by branches and leaves) remains low. In this research, a method of estimating pixel area for occluded circular masks was developed. This method involves distant pixel estimation and radius of curvature estimation. This method was evaluated via application to randomly generated occluded circular masks. The results showed that this method can decrease root mean squared error (RMSE) by an average of 87.1% for generated occluded masks. This method was then applied to realistic segmentation masks generated from in-field snapshot data as a practical application. Snapshots of peach fruits were taken with an infrared camera once every 30 min at night during the growing season. The sizes of the fruits were then estimated by the following method. First, masks were detected using Mask-RCNN, which is a popular method for instance segmentation. Next, assuming a circular fruit, the developed size estimation method was applied. Estimated results for actual images were evaluated using a logistic model tuned by an annotated mask area, which showed a maximum of 9.55% mean absolute error (MAE)-based size improvement. This size estimation method can also be applied to the field monitoring of other circular plants.

1. Introduction

Fruit size measurement is useful for determining maturity and harvest time, estimating crop yield, marketing, and estimating profit.(1) Also, peach fruit size can be a good index for water stress and fruit sweetness.(2) Optimizing growing conditions such as light and water amounts in reference to the estimated water stress can lead to improvements of taste and harvest amount. Measuring the fruit size can also lead to the optimization of operations such as fruit thinning.(3) However, the manual measurement of fruit size is expensive, and thus an automatic size estimation method should be applied.

*Corresponding author: e-mail: ryota15765@gmail.com
https://doi.org/10.18494/SAM4229

ISSN 0914-4935

Many fruit detection and size estimation methods have been developed (Table 1). Edge extraction and circle regression were used to detect occluded tomato.[4] However, this method required six hyperparameters for anonymous edge detection. These hyperparameters include values such as maximum and minimum curvatures for a normal edge, which can differ depending on camera-to-fruit distance or image resolution, and therefore, this method is difficult for practical use. Also, an estimation method for severely occluded fruits still needs improvement. In previous research, the ellipse fitting method and the allometric relationship of mango fruits were applied to RGB-D images for size estimation.[5] This method focused on non-occluded fruits, assuming that they are good representatives of all fruits. However, this can be problematic for plants with dense leaves or large size variance. Some researchers used key-point detection and strawberry shape features for size estimation.[6] Also, in a previous research study, the shape of occluded strawberries was interpolated using a two-side splitting method.[7] Similarly to these studies that made use of strawberry shape features, in this study, we aim to estimate the size of occluded circular fruits using the geometric features of a circle.

The circle hough transform (CHT) and the random sample consensus algorithm (RANSAC) are two popular methods used to extract circular objects.[8–11] These methods aim to find a circle that best fits the given image, but in the problem setting of size estimation, estimating the radius of the circle directly would be acceptable. Also, these methods adopt a voting scheme, which can lead to misdetection when there are many outlier points, which is often the case for occluded fruits. In this study, a method of directly estimating the radius, which is robust to outliers, was developed.

Analyzing images outdoors under various light conditions is also challenging. For rule-based methods, thresholds change under different light conditions, and for learning-based methods, training data should include data of all possible conditions, which can be difficult to collect.

In some previous research studies, a color index that is robust against light conditions was created to classify leaves and soil in various environments.[12] However, this method involves setting thresholds and filtering after segmentation for missed detection. There were generative adversarial network (GAN) machine learning approaches to converting light conditions for outdoor images.[13] However, this requires a large dataset for training. In this research, the SFU Grayball dataset[14] with 11346 real-world images was used for training. Collecting data under various light conditions can be difficult for practical applications. In a previous work, thermography images were used for citrus detection.[15] The authors reported that night was the

Table 1
Comparison of previous methods and experiment designs.

Reference	Target fruit	Occlusion	Light conditions	Task
(4)	Tomato	Included	Outdoor	Detection
(5)	Mango	Excluded	Outdoor	Size estimation
(6)	Strawberry	Included	Controlled	Shape and size estimation and classification
(7)	Strawberry	Included	Outdoor	Shape estimation and detection

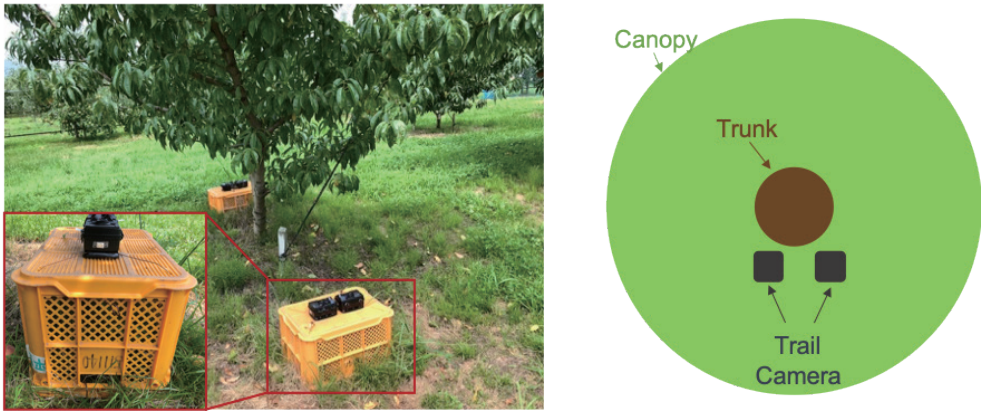
best time for detection owing to the clear temperature difference between the fruit and other tree components. Also, some researchers used a two-camera stereo vision system with controlled artificial lighting to reduce the variance of natural illumination.[16] We applied this idea to take images at night in this study. By using infrared images taken at night, the variance of light conditions can be neglected.

This research has two steps. First, a novel size estimation method for circular occluded fruits using the geometric features of a circle was proposed and tested using artificially generated occluded masks. Then, this method was applied to the predicted segmentation masks of actual peach images, which were generated by using Mask-RCNN (mask regions with convolutional neural networks). For data collection, infrared images taken at night were used to neglect the effects from various light conditions.

2. Materials and Methods

2.1 Data acquisition

The image data used in this research were collected at the Fukushima Agricultural Technology Center in Fukushima Prefecture, Japan. The peach tree used in the experiment was cultivated outdoors, as shown in Fig. 1. Images were taken during the growing season, which is about 1.5 months. The radius of the peach reaches around 8 cm at the end of the growing season. The image capture system was composed of trail cameras (Enkeeo, PH760-JP) and a box, which was fixed to the ground. The cameras were set to night mode; such cameras emit infrared (850 nm) flash and take grayscale images from the reflectance. The distance between the cameras and the fruits was 50 cm–2 m. The resolution of the cameras was 3264 × 2448 pixels. Each camera was set using a timer to take snapshots once every 30 min at night. The intrinsic parameters of the cameras were calculated beforehand by using a chessboard image and OpenCV.[17] The distortion of images was fixed using the calculated intrinsic parameters before further processing. Images from two cameras were used for this study.



(a) (b)

Fig. 1. (Color online) (a) Image of camera setting and (b) diagram of camera setting from below.

Figure 2 shows a flow chart of the method used. First, an instance segmentation model was applied to the image to predict masks of the peach fruit [(a) in Fig. 2(A)]. Then, the pixel area of occluded peach masks was estimated by our proposed method from the masks generated by segmentation models [(b) in Fig. 2(A)]. Since the shape of a peach is close to a circle, the size of occluded peaches was estimated using the shape features of circles. Finally, the estimated size was calibrated using non-occluded frames of the target peach since the shape of the peach is usually not a complete circle [(c) in Fig. 2(A)].

2.2 Instance segmentation

Recently, several models for segmentation have been developed.[18] In this research, Mask-RCNN[19] was used for the segmentation of fruits. This is one of the state-of-the-art models for object detection and instance segmentation, and is widely used in segmentation tasks.[20,21] The acquired one-channel image was converted to a three-channel image by duplicating channels to adjust the original input size of Mask-RCNN. The images were resized to 1024 × 1024 resolution, and the model was used to calculate the instance segmentation mask.

For the two cameras, 649 × 2 = 1298 infrared snapshot images were collected from May 25 to July 2, 2020 (39 days). Out of the 1298 images, 100 images of different stages were chosen and annotated. The model was trained with the 100 annotated images for 40 epochs and then was used to generate masks for all snapshot images. Figure 3 shows examples of an input infrared image and an output mask. Keras implementation (https://github.com/matterport/Mask_RCNN) was used for the calculation. Data augmentation and transfer learning were not conducted in this study. The model was trained on Tesla P4 GPU.

2.3 Pixel area estimation

The right flow chart in Fig. 2 shows the flow for the estimation of pixel area. First, with the occluded segmentation mask [Fig. 4(b)] of the fruit [Fig. 4(a)] as input, the contour of the mask

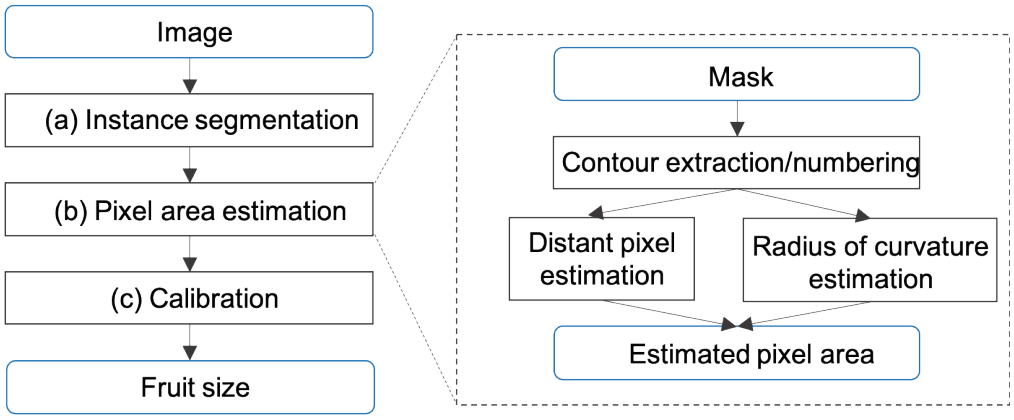
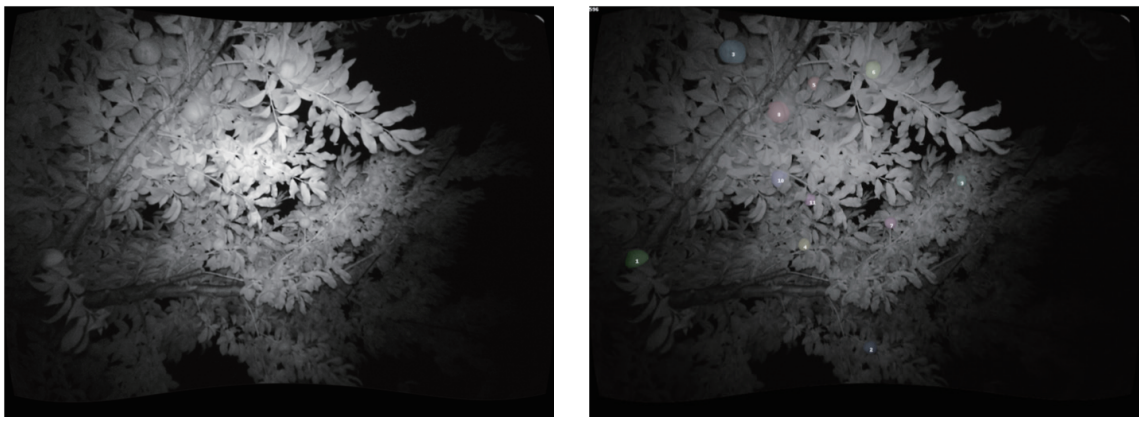


Fig. 2. (Color online) (A) Flowchart of overall size estimation and (B) flowchart of pixel area estimation.



(a) (b)

Fig. 3. (Color online) (a) Input infrared image and (b) output prediction from Mask-RCNN.

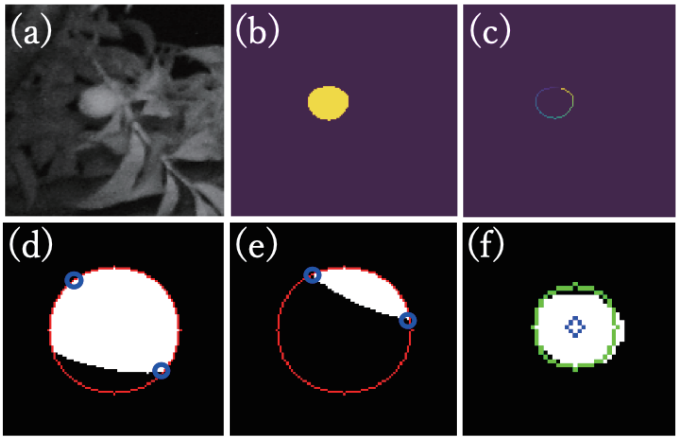


Fig. 4. (Color online) (a) Zoomed infrared image, (b) mask detected using a segmentation model, (c) numbered contour pixels, (d) and (e) examples of occluded mask (the most distant pixels are in blue and the base image contour is in red), and (f) circle detected from mask by CHT.

was extracted and numbered [Fig. 4(c); details in Sect. 2.3.1]. Next, two different methods, distant pixel estimation (see Sect. 2.3.2) and radius of curvature estimation (see Sect. 2.3.3), were used to estimate the pixel area of the mask.

The proposed method was first applied to manually generated occluded circle masks. After evaluating the generated masks, the method was applied to actual masks detected from peach images using a segmentation model.

2.3.1 Contour extraction and numbering

The contours of detected masks were extracted and the pixels in the extracted contour were numbered counterclockwise from the top pixel using the following algorithm. First, the pixel with the highest y value (and highest x value if the two were the same) was selected and

numbered '1'. Then, focusing on the surrounding 8 pixels, starting from the previous pixel position (top middle for the first pixel), the system searched 8 pixels counterclockwise until it found an existing mask pixel. That pixel was numbered '2', and then this procedure was repeated until it returned to the first pixel. The mask and numbered contour pixels are shown in Figs. 4(b) and 4(c), respectively.

2.3.2 Distant pixel estimation

This is one of the two methods for estimating pixel area from extracted contour pixels. Assuming that the mask is circular, the area of the occluded mask can be estimated using Eqs. (1) and (2), where π represents the circle ratio. This is a simple but effective method, especially when an occluded area is small. Examples of extracted distant pixels are shown in Figs. 4(d) and 4(e).

This method can estimate the size of the occluded mask when the most distant two pixels are successfully detected. However, this method underestimates the size when the occlusion is severe [(Fig. 4(e)]. Also, this method can estimate wrong values when the shape of the fruit is not circular, which is the assumption for the target fruit in this study.

$$r = \frac{\left(\text{distance of two most distant pixels}\right)}{2} \tag{1}$$

$$Area = \pi r^2 \tag{2}$$

2.3.3 Radius of curvature estimation

To improve estimation accuracy for severely occluded peaches, which is the disadvantage of distant pixel estimation (see Sect. 2.3.2), a method of estimating the radius using the curvature of the detected mask was proposed. In this method, the radius of curvature was calculated along the mask contour.

$$\theta_i = \begin{cases} \pi + a\tan\dfrac{y_i - y}{x_i - x} & [x_i < x] \\ \dfrac{3\pi}{2} & [x_i = x \; and \; y_i < y] \\ \dfrac{\pi}{2} & [x_i = x \; and \; y_i > y] \\ 2\pi + a\tan\dfrac{y_i - y}{x_i - x} & [x_i > x \; and \; y_i < y] \\ a\tan\dfrac{y_i - y}{x_i - x} & [x_i > x \; and \; y_i \geq y] \end{cases} \tag{3}$$

$$d^2 = 2r^2 - 2r^2 \cos\left(2\pi - 2\theta_1 - 2\theta_2\right) \tag{4}$$

$$r = \frac{d}{\sqrt{2}} \sqrt{\frac{1}{1 - \cos 2\left(\theta_1 + \theta_2\right)}} \tag{5}$$

$$Area = \pi r^2 \tag{6}$$

Figure 5 shows the diagram for this method. First, three points were sampled along the contour by interval I. θ_1 and θ_2 in the figure can be calculated using Eq. (3). Similar equations were also used in a previous work.[4] The radius of curvature (r) can be calculated using the cosine theorem on the triangle OAB (see Fig. 5). This procedure was iterated for three different points by sliding the extracting pixels along the contour. Then, this procedure was iterated for different interval values.

Figure 6 shows an example of the radius of curvature estimation on a mask. The middle plot shows the radius estimation along the contour. Vertical lines correspond to the same color dots in

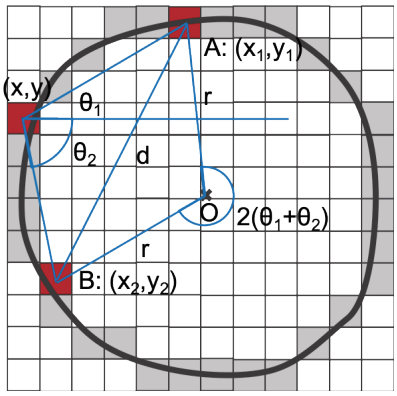
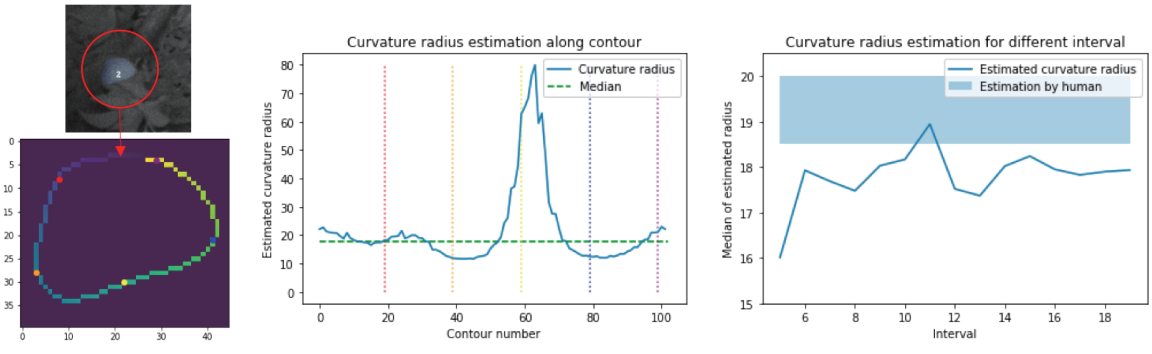


Fig. 5. (Color online) Diagram for curvature analysis with interval = 5.



(a) (b) (c)

Fig. 6. (Color online) (a, b) Example of estimated radius of curvature along contour with interval = 17. The colored vertical lines in the plot correspond to the location expressed in dots on the extracted contour with the same colors. (c) Sample plot of median radius of curvature with different interval parameters.

the mask on the left. The radius decreases when the curve is sharp and increases when the contour is loose. The median was adopted as the estimated radius to ignore noise.

As in the right plot in Fig. 6, the estimation of the radius of curvature differs by interval I. This method generally tends to underestimate the radius. To address the underestimation and noise, interval I was iterated and the radius at the 90% percentile was adopted.

This method is capable of estimating the size of severely occluded fruits [Fig. 4(e)] by using the radius of curvature. However, this method has relatively large errors compared with distant pixel estimation (see Sect. 3.1).

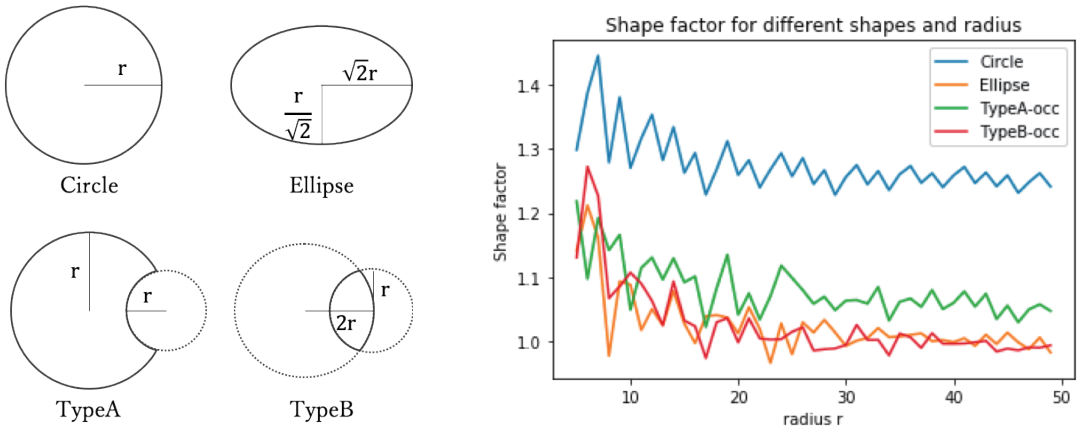
2.4 Calibration of estimated size

The area estimated by the method described assumes the fruit shape as a circle. However, the actual fruit is usually not a perfect circle, which causes overestimation by the proposed method. The estimated size was calibrated using the actual pixel area of the most non-occluded shape during measurement. Calibration was applied as follows.

- Non-occluded masks were extracted from every snapshot for each peach using circularity.
- The calibration ratio was calculated and multiplied for the estimated area.

2.4.1 Detection of non-occluded mask by circularity

Circularity (shape factor) was calculated using Eq. (7), which has been widely used since Babylonian times.[22,23] The parameters P and A represent the perimeter and area of the shape, respectively. Masks for circular fruits such as peaches tend to be close to a circle or an ellipse. This method works well for these realistically shaped masks according to the experiments shown in Fig. 7. Artificial masks were generated for the four different shapes (left part of Fig. 7) for different r values. The right plot shows circularity calculated using Eq. (7) for the generated masks. The circularity (shape factor) becomes larger and closer to 1 for circles in Euclidian



(a) (b)

Fig. 7. (Color online) (a, b) Shape factor for different realistic shapes and radii. Correspondence of shape and plot colors are shown in the legend.

spaces. However, in digital space, where the shape is discretized into pixels, the circle factor can become larger than 1.

$$\text{Circularity}\left(\text{shape factor}\right) = \frac{P^2}{4\pi A} \tag{7}$$

Masks with circularity over 90% percentile (i.e., with shapes close to a circle) were considered to be non-occluded and used for calibration.

2.4.2 Calibration by non-occluded frames

Pixel area was calculated for extracted non-occluded masks (a_1, a_2, …, a_n). By using the corresponding estimated area (b_1, b_2, …, b_n), the calibration ratio was calculated using Eq. (8). For all of the estimated areas, the calibration ratio was multiplied for calibration.

$$\text{Calibration ratio} = \frac{1}{n}\sum\frac{a_k}{b_k} \tag{8}$$

2.5 Artificial production of occluded masks for evaluation

To evaluate the size estimation method, randomly occluded masks were generated by the two conditions shown in Fig. 8. One hundred images were generated for each base radius *r*. The parameters *R* and *d* were randomly selected from the range shown in Fig. 8. Figure 9 shows some examples of generated images for different available ratios (available area/original area). Masks with an available ratio of less than 0.1 were excluded so that only realistic masks were used, since it is unlikely for segmentation models to predict masks in such occluded situations and also difficult for humans to correctly annotate that scenario.

CHT is one popular method used in such tasks.[8,9] As a method of detecting circles, it was applied to the binary mask. CHT scans all points in an image and conducts voting after conversion into (*p*, *q*, *r*) space (*p*, *q* is circle center and *r* is radius). The radius with the most votes is adopted for the most confident circle.

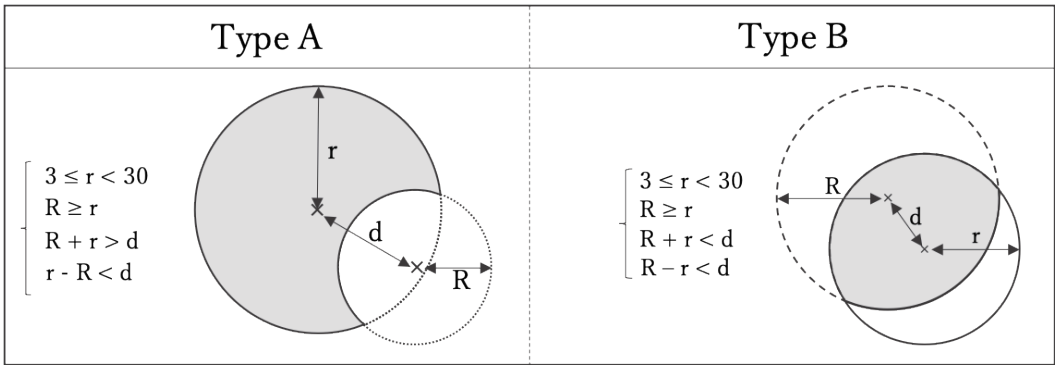


Fig. 8. (Color online) Two different methods for generating occluded masks.

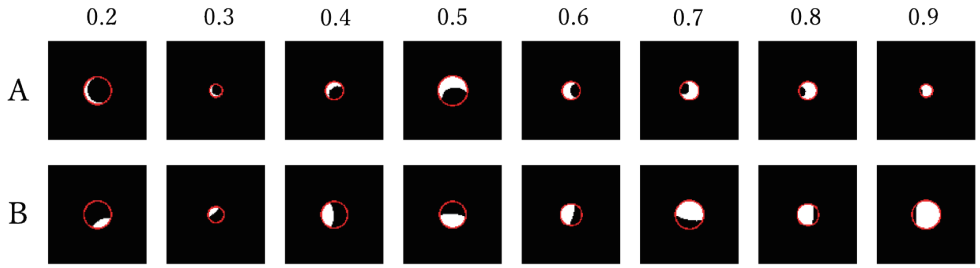


Fig. 9. (Color online) Examples of generated occluded masks from the two methods with various available ratios.

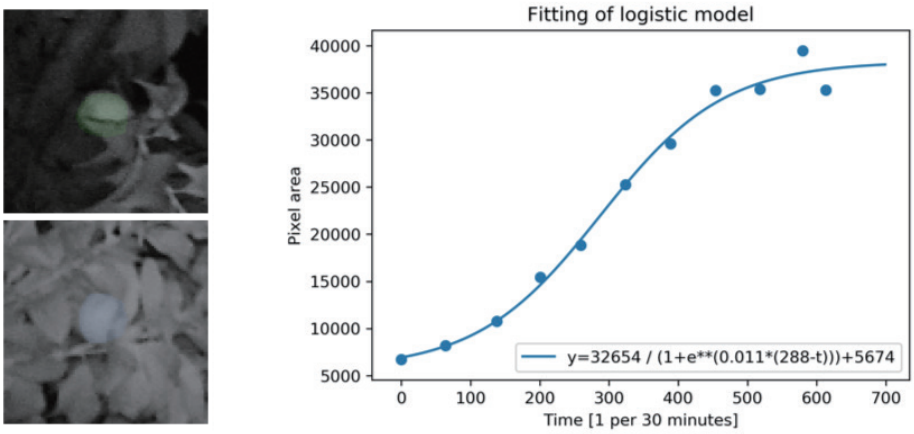
In this study, CHT was applied by using the function provided by openCV. For the parameters for CHT, the minimum and maximum radii were set to 0 and 50, respectively, to fully include the radius range of the generated masks. Parameter-1, which is the higher parameter used for Canny edge extraction, was set to 50, as decided by preliminary experiments. Parameter-2, which is the accumulator threshold used in the detection stage, was first set to 15 and decreased by iteration until a confident circle was found. The resolution of the accumulator space was set to the same value as the input image resolution. An example of circles estimated using CHT is shown in Fig. 4(f).

2.6 Evaluation of estimated size

The method of evaluating the estimated peach size was used since the actual size was not measured. Logistic models [Eq. (9)] are widely used to explain the size growth of plants such as cacao fruit,[24] cashew nuts,[25] and coffee beans[26] with satisfactory results, where y is size (normalized area in this research) and t is time, A is the asymptotic value for the fruit, k is the growth rate, B is the age at which the size reaches the reflection point, and u represents the residue.

$$y = \frac{A}{1 + e^{k(B-t)}} + u \tag{9}$$

To calculate the parameters of the logistic model, an additional manual mask annotation was made for each fruit. A series of images were equally separated into 10 sections, and the images at the edge of each section were used for annotation (11 images for each fruit, per camera). Then, the parameters of the logistic curve were optimized by the Levenberg–Marquardt method.[27] Examples of annotated masks and the tuned logistic model curve are shown in Fig. 10. Annotated masks were drawn manually to interpolate masks in an occluded situation. The logistic model curve was tuned by this flow for each fruit and used as actual data for evaluation.



(a) (b)

Fig. 10. (Color online) (a) Example of annotation. (b) Example of logistic model fitting.

3. Results and Discussion

3.1 Size estimation for manually occluded masks

Figure 11 shows the calculation results of each method for different occlusion ratios. Scatter plots show each estimation and the line plot shows the mean value for each bin: the result is divided into 10 bins according to the available ratio (e.g., 0–0.1, 0.1–0.2…). The black line indicates the area of the original occluded mask. Quantitative results are listed in Table 2. MAE(%) in Table 2 was calculated as

$$\mathrm{MAE}(\%) = \sum \frac{\mathrm{MAE}}{\text{actual area}} . \tag{10}$$

Except for the high-available-ratio area of CHT, all methods succeeded in improving the area of occluded masks. Distant pixel estimation and radius of curvature estimation improved the estimated area in both RMSE and MAE. The distant pixel estimation tended to have high accuracy, especially when the available ratio was larger than 0.4 (moving average of MAE was less than 200). This is because when any of the distant pixel pairs was left on the occluded mask, this method could estimate the mask area precisely. On the other hand, the radius of curvature estimation tended to always include some error (minimum moving average of MAE was 47.6), but the error seemed to remain low even in the low-available-ratio area (maximum moving average of MAE was 241.6). The strengths of these two methods were combined by algorithm (11), which is refered to as “ensembled” in this paper.

$$\textit{Ensembled radius} = \begin{cases} \text{distant pixel} & (\text{distant pixel} \geq \text{curvature radius}) \\ \text{curvature radius} & (\text{distant pixel} < \text{curvature radius}) \end{cases} \tag{11}$$

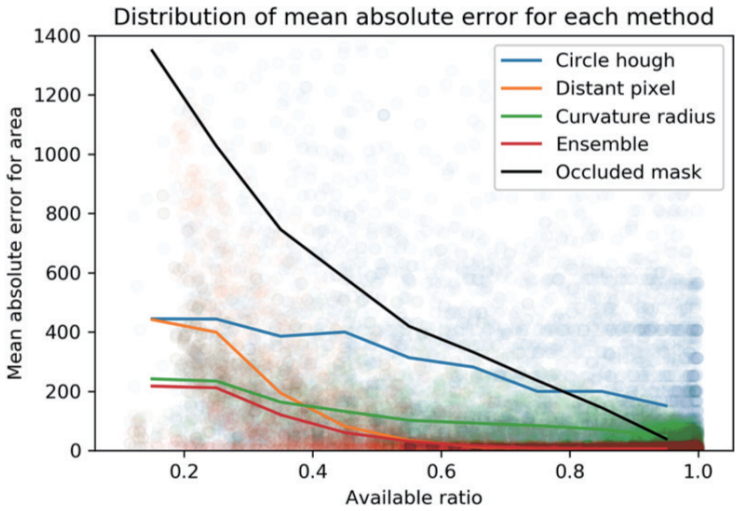


Fig. 11. (Color online) Size estimation results for generated masks (some scatterplots were trimmed for visualization).

Table 2
Evaluation of estimated RMSE, MAE, and MAE/area (%).

	Occluded mask	Circle hough	Distant pixel	Curvature radius	Ensemble
RMSE	515.2	407.6	154.9	122.4	83.9
MAE	290.8	242.9	52.5	89.9	34.0
MAE(%)	26.7	57.1	5.2	15.4	4.3

This algorithm allows the system to adopt the radius of curvature estimation results when the estimated distant pixel is larger than the estimated radius of curvature, which means that the occlusion is likely to be severe and the two most distant pixels are not likely to be included. The results shown in Fig. 11 appear to have the strength of both methods (moving average of MAE was a maximum of 217.1 and a minimum of 5.72). The combined method decreased MAE by 87.2% and RMSE by 87.1% for generated occluded masks on average. Next, the method was applied to masks for actual peach images, which can have rough contours compared with artificially generated masks.

3.2 Size estimation for peach images

The proposed method was applied to masks from three peach images with different conditions. For comparison, CHT with calibration was also applied. Results are shown in Fig. 12. Qualitative results of MAE(%) are listed in Table 3.

Peach A: This fruit was not occluded the whole time. Therefore, the original area can be considered a correct plot. The logistic model seemed to fit the peach growth curve well, since MAE(%) between the original area and the logistic model was 4.68%. MAE(%) of the combined estimation was larger than the original area by only 0.3%.

Peaches B and C: These fruits were severely occluded (available ratio was around 30 for the most occluded situation), as may be seen from the images in Fig. 12. Considering that peach size

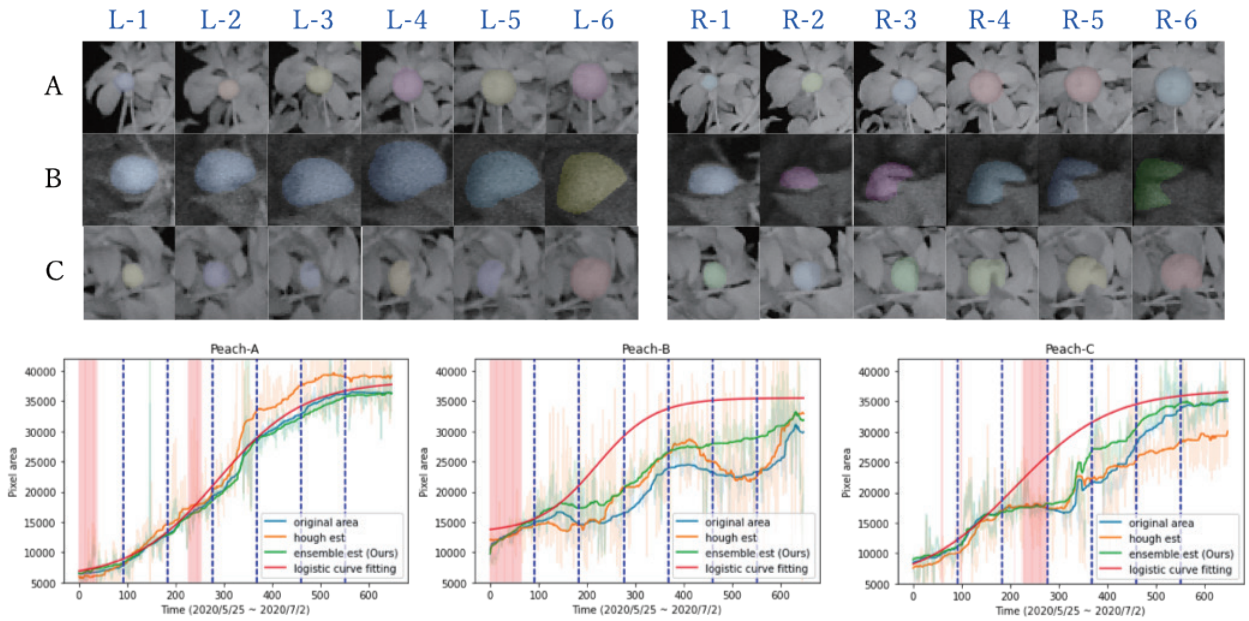


Fig. 12. (Color online) Estimation of peach masks. (top) Infrared image and masks for three peaches captured by two cameras at different times and (bottom) results of size estimation. Plots of results of original area, CHT estimation, and ensemble estimation are shown. The thick line shows the moving average by a window with a width of 95. Blue vertical lines correspond to the numbers in the image above. Pink vertical lines show the mask used for calibration.

Table 3
MAE(%) for each method and peach. This result includes the frames used for calibration.

	Original area (%)	Circle hough (%)	Ensemble (%)
Peach A	4.68	8.36	4.97
Peach B	26.99	23.99	17.04
Peach C	15.82	21.05	12.07

usually increases monotonically, decreases in the original area are likely to be caused by occlusion. From the plot, our method seems to minimize the effect of this decrease. MAE(%) was improved by 9.95% for peach B and 3.75% for peach C.

There are some remaining issues for this method. First, the prediction from the logistic model might differ from the real data. The logistic model fits non-occluded peach A by MAE(%) of 4.68%, but this could differ between fruits or conditions, and data should be taken in an actual format such as weight to be used for real application. Second, the calibration method has room for improvement. For fruits occluded the whole time, such as peach B, calibration is applied by using an area of occluded fruits. This is one of the main reasons for the underestimation of severely occluded fruits. Third, in this method, the pixel area of the mask is calculated, but the main interest of crop monitoring is the size or weight of the actual fruit. This problem can be addressed in future research by using depth information from RGB-depth images acquired by stereo cameras.[28] Also, a deep learning approach, such as occlusion region-based convolutional neural network (ORCNN), can also be used to estimate the occluded area with high accuracy.[29]

4. Conclusions

Several methods have been proposed for the detection and size estimation of fruits. However, the size estimation method for occluded situations needs improvement. Also, various light conditions for outdoor plants were challenging in previous studies.

To tackle these problems, we used infrared images, taken at night throughout the growing season to stabilize light conditions. Then, the size estimation method for occluded circular masks was developed. This method was first applied to manually generated occluded masks and yielded MAE of 34.0, which was better than that obtained by the method using CHT. This method combined with calibration was then applied to peach masks detected by Mask-RCNN. By applying the tuned logistic model for evaluation, the proposed method improved MAE(%) by 9.95% at maximum (26.99 → 17.04%). However, evaluation with actual measured data such as weight is needed for real applications. Also, the calibration method needs to be enhanced to improve underestimation.

The advantage of this method is that by using circular features to estimate the radius of the fruit, the estimated size yields acceptable accuracy for occluded fruit masks. However, this method can only be used for circular fruits, since the geometric features of a circle are used. Also, a series of images from the same location are necessary for real applications. Moreover, this method assumes that visible fruits in the image represent the size of all other fruits on the tree. This assumption can be inappropriate for fruits with different size distributions for different growing locations. In addition, only data from one tree for 1.5 months was used in this research. Validation with more diverse data can be conducted in future research.

Size estimation is important in agriculture for determining fruit maturity, deciding on harvest time, assessing crop yield, marketing, estimating profit, and controlling taste. Thus, a method of estimating the size of occluded fruits is essential, since perfectly capturing all fruits is difficult. This method improved size estimation for occluded fruits using infrared cameras in a system that is inexpensive compared with manual measurement.

References

1 S. Jenni, K. A. Stewart, G. Bourgeois, and D. C. Cloutier: J. Am. Soc. Hortic. Sci. **123** (1998) 195. https://doi.org/10.21273/JASHS.123.2.195
2 M. Genard and J. G. Huguet: Tree Physiol. **16** (1996) 407. https://doi.org/10.1093/treephys/16.4.407
3 M. Sutton, J. Doyle, D. Chavez, and A. Malladi: Horticulturae **6** (2020) 41. https://doi.org/10.3390/horticulturae6030041
4 R. Xiang, Y. Ying, and H. Jiang: 2013 6th Int. Congr. Image and Signal Processing (CISP) (Hangzhou, Dec. 2013) 713–717. https://doi.org/10.1109/CISP.2013.6745258
5 Z. Wang, K. Walsh, and B. Verma: Sensors **17** (2017) 2738. https://doi.org/10.3390/s17122738
6 L. M. Oo and N. Z. Aung: Biosyst. Eng. **170** (2018) 96. https://doi.org/10.1016/j.biosystemseng.2018.04.004
7 Y. Ge, Y. Xiong, and P. J. From: IFAC-PapersOnLine **52** (2019) 294. https://doi.org/10.1016/j.ifacol.2019.12.537
8 I. Frosio and N. A. Borghese: Pattern Recognit. **41** (2008) 1041. https://doi.org/10.1016/j.patcog.2007.08.011
9 M. T. Setiawan, S. Wibirama, and N. A. Setiawan: 2018 4th Int. Conf. Science and Technology (ICST) (Yogyakarta, Aug. 2018) 1–5. https://doi.org/10.1109/ICSTC.2018.8528286
10 M. A. Fischler and R. C. Bolles: Commun. ACM **24** (1981) 381. https://doi.org/10.1145/358669.358692
11 I. Ladrón de Guevara, J. Muñoz, O. D. de Cózar, and E. B. Blázquez: J. Math. Imaging Vision **40** (2011) 147. https://doi.org/10.1007/s10851-010-0249-8

12 K. Kirk, H. J. Andersen, A. G. Thomsen, J. R. Jørgensen, and R. N. Jørgensen: Biosyst. Eng. **104** (2009) 308. https://doi.org/10.1016/j.biosystemseng.2009.07.001
13 O. Sidorov: arXiv:1811.06604 [cs] (2019), http://arxiv.org/abs/1811.06604 (accessed 23 February 2021).
14 F. Ciurea and B. Funt: IS&T/SID Eleventh Color Imaging Conf. (2003) 160–164.
15 I. L. Argote Pedraza, J. Faber Archila Diaz, R. M. Pinto, M. Becker, and M. L. Tronco: 2019 IEEE Colombian Conf. Applications in Computational Intelligence (ColCACI) (Barranquilla, Colombia, Jun. 2019) 1–6. https://doi.org/10.1109/ColCACI.2019.8781965
16 Q. Wang, S. Nuske, M. Bergerman, and S. Singh: Experimental Robotics, Vol. 88, J. P. Desai, G. Dudek, O. Khatib, and V. Kumar, Eds. (Springer International Publishing, Heidelberg, 2013) pp. 745–758. https://doi.org/10.1007/978-3-319-00065-7_50
17 OpenCV Documentation: https://docs.opencv.org/4.x/dc/dbb/tutorial_py_calibration.html (accessed 17 March 2023).
18 S. Minaee, Y. Boykov, F. Porikli, A. Plaza, N. Kehtarnavaz, and D. Terzopoulos: arXiv:2001.05566 [cs] (2020), http://arxiv.org/abs/2001.05566 (accessed 1 March 2021).
19 K. He, G. Gkioxari, P. Dollar, and R. Girshick: 2017 IEEE Int. Conf. Computer Vision (ICCV) (Venice, Oct. 2017) 2980–2988. https://doi.org/10.1109/ICCV.2017.322
20 K. Osorio, A. Puerto, C. Pedraza, D. Jamaica, and L. Rodríguez: AgriEngineering **2** (2020) 471. https://doi.org/10.3390/agriengineering2030032
21 B. Xu, W. Wang, G. Falzon, P. Kwan, L. Guo, G. Chen, A. Tait, and D. Schneider: Comput. Electron. Agric. **171** (2020) 105300. https://doi.org/10.1016/j.compag.2020.105300
22 R. S. Montero and B. Ernesto: Int. Math. Forum **4** (2009) p. 27: 1305–1335.
23 E. R. Davies: Computer and Machine Vision: Theory, Algorithms, Practicalities (Boston: Elsevier, Amsterdam, 2012) 4th ed.
24 J. A. Muniz, M. D. S. Nascimento, T. J. Fernandes, and Universidade Federal de Lavras: Braz. Rev. Caatinga **30** (2017) 250. https://doi.org/10.1590/1983-21252017v30n128rc
25 C. A. Muianga, J. A. Muniz, M. D. S. Nascimento, T. J. Fernandes, and T. V. Savian: Rev. Bras. Frutic. **38** (2016) 22. https://doi.org/10.1590/0100-2945-295/14
26 I. F. Sousa, J. E. Kunzle Neto, J. A. Muniz, R. M. Guimarães, T. V. Savian, and F. R. Muniz: Cienc. Rural **44** (2014) 2016. https://doi.org/10.1590/0103-8478cr20131341
27 J. J. Moré: Numerical Analysis, Vol. 630, G. A. Watson, Ed. (Springer Berlin Heidelberg, Berlin, Heidelberg, 1978) pp. 105–116. https://doi.org/10.1007/BFb0067700
28 P. M. Blok, E. J. van Henten, F. K. van Evert, and G. Kootstra: Biosyst. Eng. **208** (2021) 213. https://doi.org/10.1016/j.biosystemseng.2021.06.001
29 P. Follmann, R. König, P. Härtinger, and M. Klostermann: Comput. Vision Pattern Recognit. (2018). https://doi.org/10.48550/ARXIV.1804.08864