



저작자표시-비영리-변경금지 2.0 대한민국

이용자는 아래의 조건을 따르는 경우에 한하여 자유롭게

- 이 저작물을 복제, 배포, 전송, 전시, 공연 및 방송할 수 있습니다.

다음과 같은 조건을 따라야 합니다:



저작자표시. 귀하는 원저작자를 표시하여야 합니다.



비영리. 귀하는 이 저작물을 영리 목적으로 이용할 수 없습니다.



변경금지. 귀하는 이 저작물을 개작, 변형 또는 가공할 수 없습니다.

- 귀하는, 이 저작물의 재이용이나 배포의 경우, 이 저작물에 적용된 이용허락조건을 명확하게 나타내어야 합니다.
- 저작권자로부터 별도의 허가를 받으면 이러한 조건들은 적용되지 않습니다.

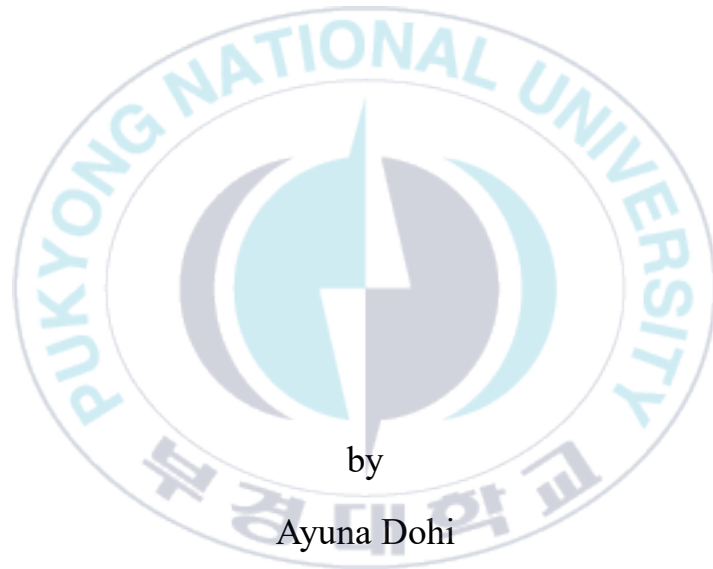
저작권법에 따른 이용자의 권리는 위의 내용에 의하여 영향을 받지 않습니다.

이것은 [이용허락규약\(Legal Code\)](#)을 이해하기 쉽게 요약한 것입니다.

[Disclaimer](#)

Thesis for Degree of Master of Engineering

Tilt-Invariant Lemon Size Estimation
Using RGB-D Camera Images



by

Ayuna Dohi

Department of Artificial Intelligence Convergence

The Graduate School

Pukyong National University

February, 2025

Tilt-Invariant Lemon Size Estimation
Using RGB-D Camera Images

RGB-D 카메라 이미지를 이용한
기울기 불변 레몬 크기 추정



Advisor: Prof. Ki-Ryong Kwon

by
Ayuna Dohi

A thesis submitted in partial fulfillment of requirements
for degree of
Master of Engineering

in Department of Artificial Intelligence Convergence, The Graduate School,

Pukyong National University

February, 2025

Tilt-Invariant Lemon Size Estimation Using RGB-D Camera Images

A Thesis
by
Ayuna Dohi

Approved by:

(Chairman)

Prof. Tae-Kuk Kim

(Member)

Prof. Xiaoyang Mao

(Member)

Prof. Ki-Ryong Kwon

February 21, 2025

Contents

I. INTRODUCTION	- 1 -
1.1 Background and Research Motivation	- 1 -
1.2 Objective of the Thesis	- 2 -
1.3 Contribution of the Thesis	- 3 -
1.4 Outline of Thesis	- 4 -
II. RELATED WORK	- 6 -
2.1 Fruit Detection	- 6 -
2.1.1 Handcrafted Methods	- 6 -
2.1.2 Deep Learning-based Methods	- 6 -
2.2 Fruit Size Estimation	- 7 -
III. PROPOSED METHOD	- 9 -
3.1 Overview	- 9 -
3.2 RGB-D Image Acquisition	- 9 -
3.3 Lemon and Tip Detection	- 11 -
3.4 Ellipse Fitting	- 12 -
3.5 Feature Extraction	- 13 -
3.6. Size Estimation Using Regression	- 16 -
IV. EXPERIMENTAL RESULTS AND DISCUSSION	- 17 -
4.1 Dataset	- 17 -
4.1.1 Dataset of Lemon Detection	- 17 -
4.1.2 Dataset of Tip Detection	- 18 -
4.1.3 Dataset of Regression Model	- 19 -
4.2. Detection of Lemon and Tip	- 20 -
4.2.1 Training Details	- 20 -
4.2.2 Evaluation Metrics for Object Detection	- 20 -
4.2.3 Experimental Results	- 21 -
4.3 Lemon Diameter Estimation	- 25 -
4.3.1 Evaluation Metrics for Regression	- 25 -
4.3.2 Comparison by Regression Model	- 25 -
4.3.3 Ablation experiment	- 26 -
4.4 comparative experiments with existing methods	- 29 -

4.4.1 Details of Subjects.....	- 30 -
4.4.2 Experimental Flow.....	- 31 -
4.4.3 Experimental Setup.....	- 32 -
4.4.4 Experimental results.....	- 34 -
4.4.5 Subjective Evaluation for Usability.....	- 39 -
V. CONCLUSION.....	- 44 -
References.....	- 45 -
List of Publications.....	- 48 -
Acknowledgment.....	- 49 -



Figures

- Fig. 1 . The current method of harvesting lemon; (a) ring for use during harvest; (b) The harvesting process.
- Fig. 2. Overview of our framework
- Fig. 3. Example of post-process of depth image; (a) original depth image; (b) filtered depth image
- Fig. 4. Detection flow of lemon and tip
- Fig. 5. Examples of $Feat_{tip}$, with (a) the tip positioned near the center and (b) the tip positioned near the edge
- Fig. 6. Images of lemon detection dataset; (a) indoor green lemon; (b) outside green lemon
- Fig. 7. Example of data augmentation for tip
- Fig. 8. The examples of lemon and tip detection
- Fig. 9. The segment errors of lemons
- Fig. 10. The absolute error between the predicted value and truth value
- Fig. 11. Experience in lemon harvesting of subjects
- Fig. 12. The flow of comparative experiment
- Fig. 13. The flow of proposed system with smartphone and edge server
- Fig. 14. Meaning of the frame and color of the estimated image
- Fig. 15. Example of difficulty in mapping lemons to each other
- Fig. 16. The processing time per image
- Fig. 17. UEQ word impression word pairs
- Fig. 18. The results of efficiency items for all experiments
- Fig. 19. The results of novelty items for all experiments

Tables

Table 1. The standard of lemon size

Table 2. The setting parameters for each regression model

Table 3. The size of collected lemon

Table 4. The results of lemon detection for validation data

Table 5. The results of tip detection for validation data

Table 6. Lemon size estimation by regression models

Table 7. The result of ablation experiment

Table 8. The experiment results in indoor-grown lemons at low positions

Table 9. The experiment results in outdoor-grown lemons at low positions

Table 10. The experiment results in outdoor-grown lemons at high positions

Table 11. The scales for attractiveness, pragmatic quality, and hedonic quality under 3 conditions

RGB-D 카메라 이미지를 이용한 기울기 불변 레몬 크기 추정

Ayuna Dohi

부 경 대 학 교 대 학 원 인공지능융합학과

요 약

과일의 크기는 시장 가치에 크게 영향을 미칩니다. 레몬의 경우 크기 등급은 단면 직경에 의해 결정되며, 특정 크기 기준을 충족하는 레몬을 수확해야 합니다. 현재의 수동 측정 방법은 금속 링을 사용하며, 이는 레몬의 품질을 저하시킬 수 있고 노동 집약적입니다. 본 연구는 RGB-D 이미지를 이용한 비접촉 방식으로 레몬의 직경을 추정하는 방법을 제안합니다. 우리의 접근 방식은 딥러닝을 사용하여 레몬과 그 끝부분을 탐지하며, 깊이 정보를 활용하고 탐지된 레몬 마스크 경계에 대한 끝부분의 위치를 바탕으로 레몬의 기울기와 관계없이 크기를 추정합니다. 실내외에서 촬영된 2,038 장의 녹색 레몬 이미지를 사용한 결과, 완전히 보이는 레몬에 대해 평균 절대 오차(MAE) 2.94mm 를 달성하였으며, 가려진 부분이 있는 경우 정확도가 떨어졌습니다. 이 연구 결과는 필드 조건에서 기울기에 상관없이 정확한 레몬 크기 측정이 가능함을 시사하며, 수확을 지원하는 유용한 도구로 활용될 수 있음을 보여줍니다.

Tilt-Invariant Lemon Size Estimation Using RGB-D Camera Images

Ayuna Dohi

Department of Artificial Intelligence Convergence, The Graduate School,

Pukyong National University

Abstract

The size of fruits significantly impacts their market value. For lemons, the size grade is determined by the cross-sectional diameter, necessitating the harvest of lemons that meet specific size criteria. Current manual measurement methods, involving metal rings, may degrade lemon quality and are labor-intensive. This study proposes a non-contact method for estimating lemon diameter using RGB-D images and deep learning. Our approach detects lemons and their tips, utilizing depth information and the position of the tip relative to the boundary of detected lemon mask to estimate size irrespective of the lemon's tilt. With 2,038 images of indoor and outdoor green lemons, our method achieved a Mean Absolute Error (MAE) of 2.94 mm for fully visible lemons, though accuracy decreased with occlusions. These findings suggest that accurate, tilt-independent lemon size measurement is feasible in field conditions, providing a valuable tool for harvest support.

I. INTRODUCTION

1.1 Background and Research Motivation

In recent years, smart agriculture has been developing, including automated agricultural works by robots, and work support based on data analysis. Various benefits are expected from smart agriculture such as a reduction of labor time, improvement of efficiency, and addressing labor shortages. In many cases, the systems determine the action and task by analyzing images captured by cameras. Until now, the systems have been restricted by numerous challenges due to the constantly changing real-world environment. However, with advancements in deep learning, it is now possible to quickly and accurately detect fruits and vegetables in real-world conditions.

Harvesting is one of the important agricultural tasks. Harvesting requires that the optimal crop be harvested based on the evaluation criteria for each crop. In the case of lemons, the size of a lemon is determined by the length of the diameter of its cross-section, and lemons that meet the shipping size are harvested. The standard of lemon is shown in table 1.

Table 1. The standard of lemon size

Grade	2S	S	M	L	2L	3L
Diameter(mm)	47	51	55	59	63	67

Lemons with a size of M or larger, defined as having a diameter of 55 mm or greater, are eligible for shipment. As shown in Fig. 1, the current method of the sizing measurement is passing a metal ring

with minimum shipping size through a lemon to ensure it meets the shipment size. Lemons that do not pass through the ring are targeted for harvesting and are being harvested by cutting them with scissors. There are 2 problems with this method.

- 1) **Degradation of lemon quality:** Since harvestable lemons do not pass through the ring, they are always in contact with it. This contact leads to surface scratches, resulting in a reduction in lemon quality.
- 2) **High workload:** The current method allows for measuring only one lemon at a time. Given that a large number of lemons are present on each tree, and the tree height exceeds 2 meters, measuring lemons located at higher positions requires the use of a stepladder, making the task particularly burdensome.



(a)



(b)

Fig. 1 . The current method of harvesting lemon; (a) ring for use during harvest; (b) The harvesting process.

1.2 Objective of the Thesis

There are various studies on size estimation of fruits and vegetables, but there is a problem that depends on the shooting orientation. In many

cases, fruit and vegetables are fitted to a circle or ellipse, and the length of the diameter of the fitted shape is used as the size estimate. Fitting methods include 2-D and 3-D fitting. In the case of 2D fitting, it allows for a shape that closely matches the region to be fitted efficiently. However, a limitation arises in that the fitted ellipse's diameter may not align with the actual diameter of the fruit or vegetable, depending on its orientation. In the case of 3D fitting, the estimated value corresponds to the diameter of the fitted shape. However, the likelihood of obtaining an accurate 3D representation is low due to insufficient or depth value errors. Moreover, processing point cloud data is computationally intensive, further complicating the approach. In real-world farm environments, network stability cannot always be guaranteed, and installing equipment with high processing capabilities is often challenging. Under these conditions, 3D reconstruction is not suitable for real-time, high-speed processing. Therefore, in this study, the orientation of the lemon was calculated by detecting its tip to estimate the diameter while accounting for the lemon's inclination, using 2D ellipse fitting.

1.3 Contribution of the Thesis

To solve the problems of degradation of lemon quality and high workload, we propose to use images to estimate the length of the lemon diameter as the size of the lemon. The use of images eliminates the drawbacks of using rings, which the contact with the lemon and the fact that only one lemon can be measured at a time. In this research, RGB-D images, consisting of paired RGB and depth images, are utilized. The

use of depth images in addition to RGB images has the advantage that the distance from the camera to the lemon can also be measured and 3-dimensional information can be obtained. In the diameter estimation method, the position of the diameter is influenced by the orientation of the lemon in the image. Given that the shape of a lemon resembles an ellipse, we can approach this problem using ellipse fitting techniques. For lemons photographed perpendicular to the camera, the position of the actual diameter closely aligns with that of the ellipse's diameter. However, when the lemon is photographed at an angle, the actual diameter becomes misaligned with the ellipse's minor axis. Therefore, simply calculating the length of the minor axis from ellipse fitting is insufficient to estimate the diameter accurately when considering the tilt of the lemon. In this research, the orientation of the lemon is estimated by detecting its tip. The assumption is that the lemon is more inclined when the tip is located centrally, and more perpendicular to the camera when the tip is positioned near the edge. The proposed method involves detecting both the lemon and its tip from the RGB image using a deep learning-based object detection model and estimating the lemon's diameter by predicting five features derived from the depth image and detection results using a regression model. The regression model is trained on features representing the lemon's slope, obtained from the tip detection, enabling tilt-invariant diameter estimation.

1.4 Outline of Thesis

The thesis is organized as follows:

- Chapter 1: The introduction part explains background and motivation of this thesis, point out objective of the thesis, and summary contributions of the thesis.
- Chapter 2: The related work part discusses the conventional and state-of-the-art methods for object detection, vegetables and fruits sizing.
- Chapter 3: In this chapter, we present the proposed method. This section discusses in detail the part of our technique such as: overall of proposed method, lemon and tip detection, calculating features for regression, regression method.
- Chapter 4: In this chapter, we present the experiment results obtained using our methods and discuss our performance evaluation.
- Chapter 5: The final chapter contains the conclusion about our research and a brief consideration concerning future works.

II. RELATED WORK

2.1 Fruit Detection

In the field of smart agriculture, fruit detection methods can be divided into 2 categories: 1) handcrafted methods based on human-defined features, and 2) deep learning-based methods. Handcrafted methods often use color thresholds to detect fruit.

2.1.1 Handcrafted Methods

For example, [3] successfully separated the background and fruit using Otsu's threshold[12] with three color spaces. Otsu's threshold[12] is a method to find a threshold value that separates the foreground and background. It minimizes the variance within each group while maximizing the value between classes. Reference [4] attempted detection based on changes in brightness values. Handcrafted detection methods are limited by their sensitivity to image noise and color variations of the fruit. For instance, they behave unexpectedly when there is no clear color difference between the background and the fruit. In the case of green lemons, the background often contains many leaves, and since the green lemon fruits are also green, it can be challenging to distinguish them. In real field conditions, variations in lighting due to sunlight and occlusions like leaves can cause errors easily in fruit detection.

2.1.2 Deep Learning-based Methods

Deep learning-based fruit detection commonly uses object detection or instance segmentation. Deep learning models can be optimized with training data and are robust to noise, making them suitable for complex

field environments. Object detection or instance segmentation methods are commonly used in detecting fruits in images. Object detection involves identifying rectangular bounding boxes around detected objects and assigning class probabilities that represent the likelihood of the objects being a specific class. On the other hand, instance segmentation can detect objects at the pixel level in addition to providing bounding boxes and class probabilities. Reference [5] improved Mask R-CNN [6], a type of instance segmentation model, for apple detection, showing high detection accuracy even with shadows and occlusions. In recent years, models with a structure You Only Look Once (YOLO) have been developed, enabling high-speed execution and being utilized for various tasks. YOLO achieves fast processing by simultaneously detecting the class and location of objects within an image. There are many models in the YOLO series, but among them, YOLACT[7], YOLOv8[8], and YOLOv11[9] stand out as models capable of instance segmentation. Recently, YOLOv8 [8] has made real-time instance segmentation possible with high accuracy. Reference [10] used YOLOv8 to detect green apples.

2.2 Fruit Size Estimation

For size estimation, detection results are often fitted to shapes like circles or ellipses. Reference [11] estimated mango size by fitting a mask image to an ellipse after detection. Another approach involves combining RGB and depth images to create a 3D shape of the fruit for size estimation. For green apples, [10] applied the least-squares method and differential equation in order to fit an ellipsoid. When using only

2D images, fruits often appear tilted depending on the shooting position. This tilt can make it difficult to estimate the correct size based solely on the fitted shape. 3D methods tend to be more accurate but can suffer from increased computation time and inaccuracies due to insufficient depth information.

This study proposes a lightweight size estimation method for lemons that calculates the fruit's tilt from images, using detected features and depth information to provide tilt-independent size estimation.



III. PROPOSED METHOD

3.1 Overview

The overview of our framework is given in Fig. 1. First, acquiring RGB-D images((a), (b) in Fig. 2) and complementing the error value for depth image((c) in Fig. 2). Then, detecting lemon((d) in Fig. 2) and tip((f) in Fig. 2) utilizing deep learning models. Subsequently, elliptical fitting to enhance comprehension of the morphology of a lemon((e) in Fig. 2). After extracting features from the results of detection and depth values, compute the actual lemon size by a regression model((g) in Fig. 2). We will explain each process in the following sections.

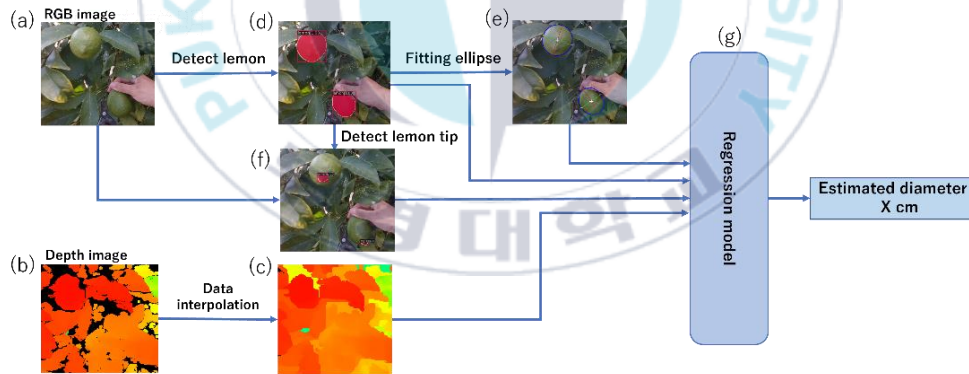


Fig. 2. Overview of our framework

3.2 RGB-D Image Acquisition

We used Intel RealSenseD415, D435i, and D435f RGB-D cameras to capture RGB and depth images. The RealSense series uses stereo vision and Infrared Rays(IR) pattern to obtain depth image. IR pattern from the infrared projector indicates invisible static infrared patterns,

utilized for the improvement of depth accuracy. We utilized the RealSense library provided by Intel to acquire and post-process RGB-D images. The acquired depth images have 2 issues: 1) The field of view (FOV) of the RGB and depth images differs due to the different positions of the cameras, making it impossible to reference them as a pair. 2) The captured environment potentially includes missing values and outliers, especially around objects whose depth values are unclear. To address the FOV issue, we aligned the images using the internal and external parameters of the cameras. For the outliers, we applied smoothing and filling processes. Smoothing was performed using the spatial edge-preserving filter provided by RealSense official[13], calculated as follows:

$$S_t = \begin{cases} Y_1 & t = 1 \\ \alpha Y_t + (1 - \alpha)S_{t-1} & t > 1 \text{ and } \Delta = |S_t - S_{t-1}| < \delta \\ Y_t & t > 1 \text{ and } \Delta = |S_t - S_{t-1}| > \delta \end{cases} \quad (1)$$

Y_t is the recorded depth and S_t is the calculated depth value at any time period t . α and δ represent the number of iterations, the degree of weighting decrease, and the step-size boundary, respectively. When the difference between the measured and last calculated depth exceeds the depth threshold set by δ , set the newest depth value. It applies to hole-filling for missing values, which is calculated by referencing the maximum value among the five adjacent pixels: above, below, left, and right. Fig. 3 shows the result of post-process of depth image.

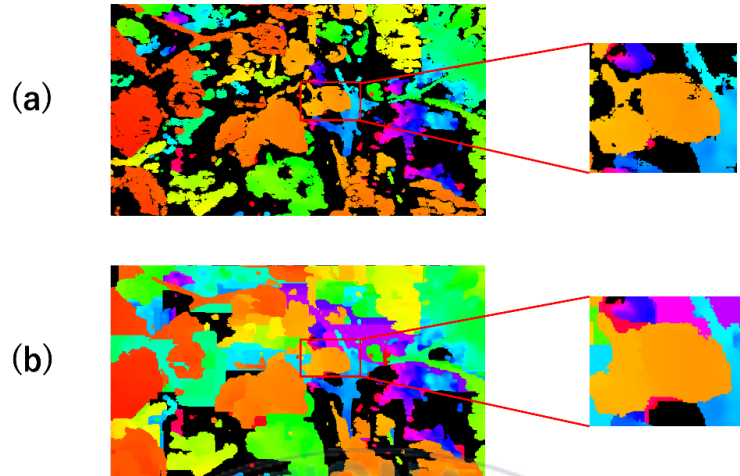


Fig. 3. Example of post-process of depth image; (a) original depth image; (b) filtered depth image

3.3 Lemon and Tip Detection

To achieve effective performance in estimating lemon diameter in the real field, both accuracy and execution time are crucial. In fruit size estimation, identifying the fruit's area is an important clue for understanding its shape and size. Therefore, we use an instance segmentation model, and this study employs the anchor-based Mask R-CNN and the anchor-free Yolov8 methods. Detection is carried out in 2 stages: detecting the lemon and then the lemon's tip, as illustrated in Fig. 4. As shown in Fig. 4, first, we detect lemons from the original image. Since the lemon tip is a very small area relative to the whole image, it is difficult to detect directly from the original image because there are similar features all over the place. Therefore, we crop the original image based on the bounding box and then detect the tip for each lemon. Considering the orientation of the lemon, we classify the lemon's tip. However, since the tip also occupies a small area relative

to the entire lemon, similar features might appear elsewhere on the lemon. To improve detection rates, we divide the tips into 2 classes: 'side tip' (on the lemon's outline) and 'center tip' (inside the lemon). If a lemon has multiple tips, we select the one with the highest class probability. As a result, the corresponding tip is identified for lemons where the tip is detected.

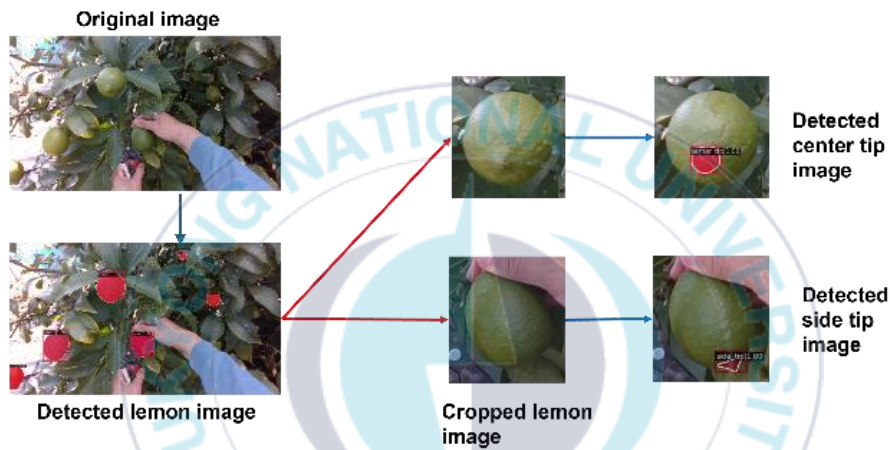


Fig. 4. Detection flow of lemon and tip

3.4 Ellipse Fitting

We approximate the shape of a lemon's outer contour as an ellipse, using the detection results of the lemon. Lemons frequently overlap each other in actual farms. It is highly likely that multiple lemons will be approximated as a single in the case of overlapped lemons. Thus, ellipse fitting method for whole image does not work effectively. To approximate the ellipse for each lemon individually, we crop the image based on the bounding box from the detection results. Using the mask information from the detection results, separating the lemon from the background. Since the mask area may be separated as several regions in some detection results, we perform a morphological closing

operation with a 7×7 kernel to connect these areas. Then, we fit an ellipse to the mask image using the least squares method. This process provides us with the ellipse's center coordinates, the lengths of the short and long axes, and the rotation angle.

3.5 Feature Extraction

We calculate features to input into the regression model using depth images, lemon detection results, tip detection results, and ellipse fitting results. The five features used are listed below.

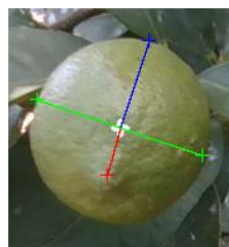
1. $Feat_{area}$: the number of pixels in the mask.
2. $Feat_{depth}$: the depth value at the center of the ellipse.
3. $Feat_{tip}$: the ratio of the length from the tip to ellipse center to the length of the major axis of the ellipse.
4. $Feat_{depth_{diff}}$: absolute difference in depth values at the endpoints of the ellipse's minor axis.
5. $Feat_{short_{axis}}$: estimated diameter of the lemon using the length of the ellipse's minor axis in real world units.

$Feat_{area}$ represents the area of the lemon in the image, we use the total number of pixels in the lemon mask obtained from detection result. $Feat_{depth}$ represents the depth of lemon center, which is used the depth value of the center of ellipse as center depth. The tilt of the lemon in the depth direction can significantly affect its shape and diameter appearing on 2D image. Therefore, we assume that the center of the fitted ellipse is always at the center of the lemon in the image,

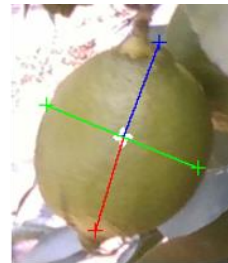
regardless of its tilt. Then determine $Feat_{tip}$, the lemon's orientation based on the relationship between the ellipse center and the position of the tip. We considered the tilt of the lemon $Feat_{tip}$ based on the relationship between the center of the ellipse and the tips, assuming that the closer the 2 points are, the greater the tilt. Specifically, we used the ratio of the distance from the tip to the center of the ellipse to the length of the major axis of the ellipse and calculated it using (2).

$$Feat_{tip} = \frac{d_{tip}}{long\ axis} \quad (2)$$

d_{tip} represents the number of pixels from the center of the ellipse to the center of the detected tip's bounding box, and $long\ axis$ is the number of pixels of the ellipse's major axis. $Feat_{tip}$ ranges from 0.0 to 1.0. If it exceeds 1.0 or the tip is not detected, setting to 1.0. A value close to 0.0 means the tip is near the center, indicating a large tilt, while a value close to 1.0 means the tip is near the edge of the lemon, indicating a small tilt. A calculation example is shown in Fig. 5.



$Feat_{tip} = 0.53$
(a)



$Feat_{tip} = 0.97$
(b)

Fig. 5. Examples of $Feat_{tip}$, with (a) the tip positioned near the center and (b) the tip positioned near the edge

The depth values of the endpoints of the minor axis of the ellipse change depending on the tilt and orientation of the lemon. $Feat_{depth_diff}$, the absolute difference in depth values between the 2 endpoints of the minor axis is used as one of the cues to predict the rotation of the lemon. The minor axis of the ellipse $Feat_{short_axis}$ is considered as the diameter of the lemon, and the length of the minor axis in the real world is determined using the ellipse fitting results and the depth image. Since depth values tend to be unstable near the edges of the object, the accuracy of the depth values at the endpoints of the minor axis becomes unreliable. Therefore, new depth values are selected from the vicinity of the endpoints. Specifically, the difference between the depth value of the ellipse center and the depth values near the endpoints is calculated, and the depth values are replaced with those that have a difference below a threshold. In this case, the threshold is set to 100 mm based on the maximum diameter of the collected lemon. The search range is expanded by drawing a circle around the endpoints, and the first depth value that is below the threshold is adopted. Next, using the image coordinates of the minor axis endpoints, the selected depth values, and the camera's internal parameters, the 3D coordinates of the lemon relative to the camera, i.e., the camera coordinate system, are determined using (3).

$$\begin{cases} x_{camera} = \frac{x_{img} - c_x}{f_x} d \\ y_{camera} = \frac{y_{img} - c_y}{f_y} d \\ z_{camera} = d \end{cases} \quad (3)$$

x_{img} and y_{img} represent the coordinates on the image. c_x and c_y are the optical center coordinates, and f_x and f_y are the focal lengths.

Next, using the 2 endpoints transformed into camera coordinates, the Euclidean distance between the 2 points is calculated using (4)

$$Feat_{short\ axis} = \sqrt{(x_A - x_B)^2 + (y_A - y_B)^2} \quad (4)$$

where (x_A, y_A) and (x_B, y_B) denote the x and y coordinates in the 3D camera coordinate system for the left and right endpoints, respectively.

3.6. Size Estimation Using Regression

We input the extracted features into a regression model to estimate the diameter of the lemons. In this study, we evaluated 4 regression models: Lasso [14], Ridge [15], Elastic Net [16], and Random Forest [17] chosen for their lightweight computational requirements, which enable quick execution. The setting hyperparameters for each model are shown in Table 2. Lasso, Ridge, and Elastic Net were set alpha parameter, and Random Forest was set 2 parameters, $n_estimators$ and max_depth . The weights obtained through training are utilized to produce the final size estimation results.

Table 2. The setting parameters for each regression model

	Parameter	Values
Lasso	alpha	[0.01, 0.05, 0.1, 0.25, 0.5, 1.0, 5.0, 10]
Ridge	alpha	[0.01, 0.05, 0.1, 0.25, 0.5, 1.0, 5.0, 10]
Elastic Net	alpha	[0.01, 0.05, 0.1, 0.25, 0.5, 1.0, 5.0, 10]
Random Forest	$n_estimators$	[4, 8, 16, 32, 64, 128, 256, 512]
	max_depth	[8, 16, 32, 64, 128]

IV. EXPERIMENTAL RESULTS AND DISCUSSION

In this section, we describe the dataset used, as well as the detection and size estimation results, and discuss the experiment result.

4.1 Dataset

In this experiment, we created three custom datasets of green lemon: lemon detection, tip detection, and size estimation. The lemon dataset is used to identify the lemon regions in the images, and the tip dataset is used to estimate the orientation of the lemons. The dataset of size estimation is used to train and test for regression model of lemon size estimation. We collected green lemon RGB-D images of indoor and outdoor cultivated lemons taken over 2 years, from 2022 to 2023, to create these datasets. All lemons were at the harvest-ready stage. The cameras used were the RealSense series D415, D435i, and D435f, and both RGB and depth images were captured at a resolution of 1280×720 . The sizes of the lemons collected are shown in Table 3.

Table 3. The size of collected lemon

	Mean	Min	Max	Standard deviation
Diameter(mm)	5.28	2.90	7.00	0.59

4.1.1 Dataset of Lemon Detection

An example of the images used in the datasets is shown in Fig. 6. For the lemon detection, a total of 2,028 images were used for training and 420 images were used for validation.



Fig. 6. Images of lemon detection dataset; (a) indoor green lemon;
(b) outside green lemon

4.1.2 Dataset of Tip Detection

For the tip detection dataset, 1,894 images of lemons cropped based on bounding box were utilized. All images included bounding box and mask information. A total of 1,488 lemon cropped images were used as training data, with data augmentation applied through rotation and brightness variation. The reason for these augmentations is to account for changes in lighting conditions and variations in lemon orientation that occur in real-world farm environments. In this research, image rotation was applied within the range of $-90^\circ \leq \theta \leq 90^\circ$, and brightness was adjusted using the following equation (5).

$$dst(x, y) = \alpha * src(x, y) \quad (5)$$

$src(x,y)$ indicates the RGB value in the input image (x,y) coordinates. $dst(x,y)$ indicates the output RGB value. if $dst(x,y)$ is greater than 255, the value is changed to 255. In this study, the change was made within the range of $0.6 \leq \alpha \leq 1.6$. Fig. 7 shows an example of data expansion. 2,976 images, including images applied data augmentation, were used as training data. For the validation data, 406 lemon images that were not used as training data were used.



Fig. 7. Example of data augmentation for tip

4.1.3 Dataset of Regression Model

For the purpose of training the size estimation regression model, we used a total of 1,663 cropped images of lemons, which were selected

from the images used for tip detection, and an additional 744 cropped images of lemons.

4.2. Detection of Lemon and Tip

4.2.1 Training Details

All the experiments were implemented on a workstation with Intel(R) Xeon(R) Gold 6326 @2.90 GHz CPU, 64.0GB and NVIDIA RTX A6000 GPU. The deep neural networks for lemon and tip detection were trained with a maximum iteration of 100 and a batch size of 16. The models Mask R-CNN[6], Cascade Mask R-CNN[18], YOLOv8[8], YOLOv11[9] were utilized. YOLOv8 and YOLOv11 have 5 different types according to the amounts of trainable parameters.

4.2.2 Evaluation Metrics for Object Detection

We used the following evaluation metrics to assess the model performance: mean Average Precision(mAP), and Floating Point Operations Per Second(Flops). mAP indicates the area of Precision-Recall curve, the value range between 0.0 ~ 1.0. precision and recall are calculated by (6) and (7).

$$Precision = \frac{True\ Positive}{True\ Positive + Positive} \quad (6)$$

$$Recall = \frac{True\ Positive}{True\ Positive + False\ Negatives} \quad (7)$$

Recall means the ability of a model to identify all relevant instances in the dataset. Precision means the accuracy of the positive predictions made by the model. It is the ratio of correctly identified positive samples (true positives) to all predicted positives (true positives + false

positives). It is the ratio of correctly identified positive samples (true positives) to the total actual positives (true positives + false negatives). The detection rate improves as the value increases. FLOPS refers to the number of floating point operations per second, with lower values indicating fewer computational resources required.

4.2.3 Experimental Results

The segmentation results of lemon and tip are shown in Table 4 and Table 5, respectively.

Table 4. The results of lemon detection for validation data

Model	Backbone	mAP_{50}	mAP_{75}	FLOPs
Mask R-CNN	Resnet50	0.829	0.733	259 G
Cascade Mask R-CNN	Resnet50	0.805	0.730	1780G
Cascade Mask R-CNN	Resnet101	0.816	0.740	1850G
YOACT	Resnet50	0.766	0.620	61.4G
YOACT	Resnet101	0.775	0.636	84.7G
YOLOv8n	custom CSPDarknet53	0.760	0.583	12.0G
YOLOv8s	custom CSPDarknet53	0.687	0.595	42.4G
YOLOv8m	custom CSPDarknet53	0.688	0.598	110.0G
YOLOv8l	custom CSPDarknet53	0.674	0.589	220.1G
YOLOv8x	custom CSPDarknet53	0.797	0.614	313.0G
YOLOv11n	Convolution layer and C3k2 block	0.778	0.576	10.2G
YOLOv11s	Convolution layer and C3k2 block	0.791	0.570	35.3G
YOLOv11m	Convolution layer and C3k2 block	0.806	0.611	123.0G
YOLOv11l	Convolution layer and C3k2 block	0.794	0.589	141.9G
YOLOv11x	Convolution layer and C3k2 block	0.788	0.604	319.7G

Table 5. The results of tip detection for validation data

Model	Backbone	mAP_{50}			mAP_{75}			FLOPs
		Side	Center	All	Side	Center	All	
Mask R-CNN	Resnet50	0.827	0.673	0.750	0.123	0.186	0.154	209.0G
Cascade Mask R-CNN	Resnet50	0.785	0.629	0.707	0.117	0.113	0.115	1780.0G
Cascade Mask R-CNN	Resnet101	0.822	0.626	0.724	0.143	0.186	0.164	1785.0G
YOACT	Resnet50	0.861	0.733	0.799	0.116	0.160	0.147	61.5G
YOACT	Resnet101	0.860	0.769	0.814	0.123	0.283	0.132	84.8G
YOLOv8n	custom CSPDarknet53	0.877	0.717	0.797	0.219	0.270	0.244	10.7G
YOLOv8s	custom CSPDarknet53	0.857	0.793	0.825	0.218	0.292	0.255	37.3G
YOLOv8m	custom CSPDarknet53	0.881	0.746	0.814	0.226	0.250	0.238	98.7G
YOLOv8l	custom CSPDarknet53	0.879	0.768	0.824	0.249	0.228	0.239	200.5G
YOLOv8x	custom CSPDarknet53	0.818	0.818	0.818	0.184	0.337	0.261	313.0G
YOLOv11n	Convolution layer and C3k2 block	0.841	0.822	0.831	0.238	0.241	0.239	10.2G
YOLOv11s	Convolution layer and C3k2 block	0.880	0.816	0.848	0.214	0.345	0.280	35.3G
YOLOv11m	Convolution layer and C3k2 block	0.856	0.796	0.826	0.231	0.295	0.263	123.0G
YOLOv11l	Convolution layer and C3k2 block	0.795	0.780	0.787	0.204	0.274	0.239	141.9G
YOLOv11x	Convolution layer and C3k2 block	0.830	0.732	0.781	0.210	0.174	0.192	318.5G

As shown in Table 4, Mask R-CNN outperforms all models in mAP_{50} and Cascade Mask R-CNN is the highest accuracy in mAP_{75} for lemon detection. On the other hands, as shown in Table 5, YOLOv11s achieves better results for tip detection. In the case of tip detection, the mAP

decreased significantly from mAP_{50} to mAP_{75} in all models. This drop in mAP can be attributed to the unclear contours of the tips, leading to deviations between the detected results and the ground truth. Nonetheless, as tip detection aims to capture their relative positions with respect to the lemon center, the mAP_{50} is considered sufficient. When focusing on computational cost, the FLOPs between Mask R-CNN series and YOLO series show significant differences when detecting lemons and tips. Mask R-CNN achieves Flops of 259G for lemon detection, whereas YOLOv8n reaches Flops of 12G. This indicates that YOLOv8n is much less computationally intensive and faster than masked R-CNNs. The predicted images of lemon and tip are shown in Fig. 8. The superior accuracy of Mask R-CNN can be attributed to its higher detection for partially occluded lemons, as shown in Fig. 9(a), and to the frequent misclassification of leaves as lemons by YOLOv8n as shown in Fig. 9(b). This indicates that Mask R-CNN is more adept at distinguishing lemons from other objects, even under challenging conditions. When considering both speed and accuracy as the primary criteria for model selection, YOLOv11s emerges as a more suitable choice. This is largely due to its ability to achieve a balance between computational efficiency and performance metrics. The model demonstrates relatively high accuracy, with mAP_{50} values of 0.791 for detecting lemons and 0.880 for identifying tips. These results indicate

that YOLOv11s not only performs well in terms of precision but also maintains an edge in real-time applications, making it a practical solution for tasks where both speed and accuracy are crucial.



Fig. 8. The examples of lemon and tip detection

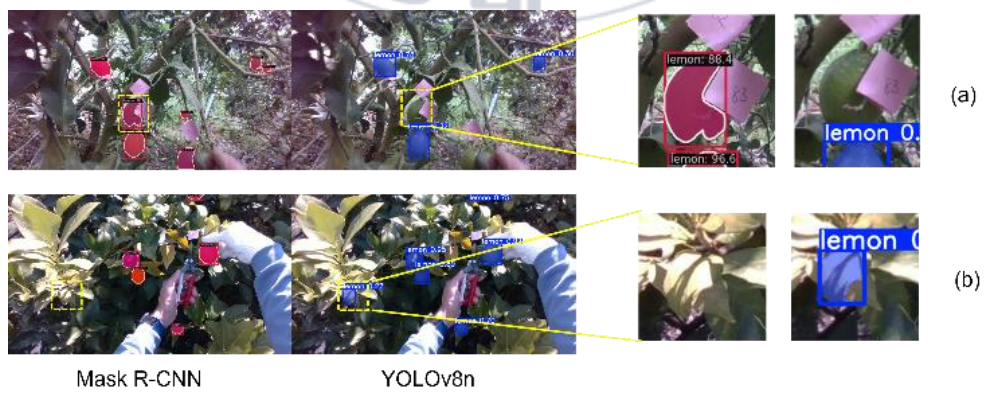


Fig. 9. The segment errors of lemons

4.3 Lemon Diameter Estimation

4.3.1 Evaluation Metrics for Regression

To assess the performance of these models, we used three evaluation metrics: mean absolute Mean Absolute Error(MAE), Mean Squared Error(MSE), and Root Mean Squared Error(RMSE).

$$MAE = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i| \quad (8)$$

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2 \quad (9)$$

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2} \quad (10)$$

where n indicates the number of data, y_i is the prediction value and \hat{y}_i is the true value. All calculations are based on the error between the ground truth and the predicted values. However, the way the error is measured whether using the absolute value, the squared value, or the square root affects how much larger errors impact the results.

4.3.2 Comparison by Regression Model

We evaluated lemon size estimation using four regression models: Lasso[14], Ridge[15], Elastic Net[16], and Random Forest[17]. These models use five calculated lemon features as input, with the first three being linear models that use different regularization methods to prevent overfitting, while Random Forest uses multiple decision trees to generate results.

Table 6. Lemon size estimation by regression models

	MAE (mm)	MSE(mm)	RMSE(mm)
Lasso	3.72	48.78	6.98
Ridge	3.87	97.09	9.85
Elastic Net	3.73	42.52	6.52
Random Forest	2.94	15.12	3.89

The results of size estimation using YOLOv8n for the validation data are shown in Table 6. The Random Forest outperformed other models in all three evaluation metrics for size estimation on the test data. The other three regression models (Lasso, Ridge, and Elastic Net) showed similar MAE values. While Elastic Net had the lowest RMSE of 42.52 among these three, it was still more than twice that of Random Forest. This suggests limitations in linear regression models for this task. MSE and RMSE tend to increase significantly with larger errors. Therefore, Random Forest's superior performance in these metrics indicates that it not only has lower average errors but also fewer extreme errors compared to other models.

4.3.3 Ablation experiment

We divided the five features used for regression into three elements to analyze how each feature impacts the accuracy of diameter estimation. This approach allows us to better understand the contribution of each feature to the overall regression performance. The five features can be categorized into 3 elements: *length*, *scale*, and *rotation*.

- 1) Length refers to $Feat_{short_axis}$, which is the diameter measured from the ellipse.
- 2) Scale refers to $Feat_{area}$ and $Feat_{depth}$, representing the relative size of the lemon in the image from the camera's perspective.
- 3) Rotation refers to the lemon's orientation and tilt, estimated using $Feat_{tip}$ and $Feat_{depth_diff}$.

We compared four conditions: baseline using only the short axis, baseline with added rotation information, baseline with added scale information, and using all features together. Detection in all cases was performed using YOLOv8. The predicted diameter in the short axis case was based on values calculated during feature extraction, while for the other three conditions, regression was performed using a Random Forest model for training and testing. Table 7 shows how scale and rotation contribute to size estimation, using shot axis as the baseline for the validation data. We evaluate this contribution using 3 metrics: MAE, MSE, and RMSE. Furthermore, the standard deviation represents the variation in the absolute error between the predicted values and the true values.

Table 7. The result of ablation experiment

	Max(mm)	MAE (mm)	MSE(mm)	RMSE	Standard deviation
Short Axis	414.02	13.78	1888.82	43.46	41.28
+ Rotation	15.39	4.44	30.08	5.48	3.23
+ Scale	17.73	3.12	16.40	4.05	2.59
Full	16.88	2.94	15.12	3.89	2.55

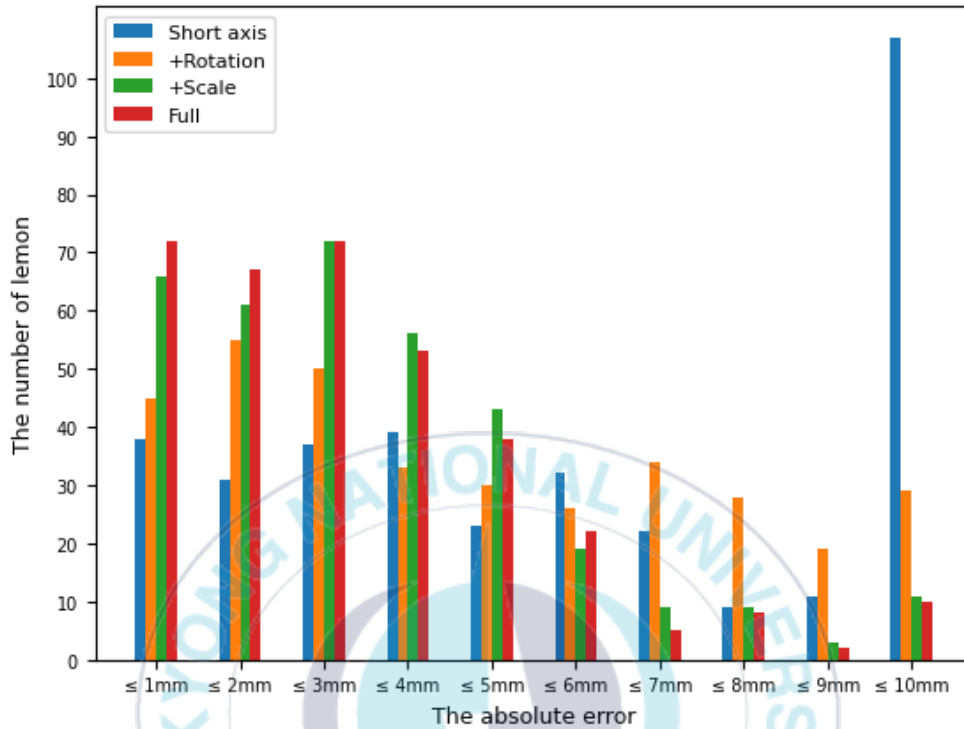


Fig. 10. The absolute error between the predicted value and truth value

The improvements in size estimation accuracy when combining different features. Using only the short axis length of the ellipse resulted in an MAE of 13.78 and an RMSE of 43.46. MSE, in particular, has a significantly larger value of 1888.82. Fig. 10 shows the absolute error between the predicted value and the truth value for each feature combination. From Fig. 10, it is clear that when using only the short axis feature, there are a large number of lemons with errors greater than 10 mm. As a result, MSE and RMSE, which are sensitive to large errors, reached high values, compared to other methods. When size features were added to the scale features, MAE and RMSE improved dramatically to 3.12 and 4.05, respectively when compared with the case of only the short axis length. Similarly, adding rotation features to the feature of short axis length improved all metrics. The initial

maximum error of $\pm 414.02\text{mm}$ likely occurred when depth values at the ellipse's short axis endpoints were incorrectly captured, leading to large depth differences and consequently, significant size estimation errors. The regression model appears to have mitigated these extreme errors. The highest accuracy was achieved when all five features were used together. The improvement seen when using all features can be explained by the complex nature of lemon appearance in images. For instance, when a lemon rotates in place, its area in the image changes, which could mislead size estimates based solely on area. However, by incorporating rotation-related features, the model can better account for these orientation-induced size variations, leading to enhanced accuracy. This approach allows the model to adapt to lemon orientations, resulting in a more robust size estimation system.

On the other hand, when lemons are obscured by leaves or other lemons, the features input for regression may not be correctly captured, leading to decreased accuracy. In the case of area estimation, using the mask image of the lemon detection result means that information about the obscured parts is lost. Additionally, if the tip is hidden, it cannot be detected. Therefore, a future challenge is to improve accuracy in situations where lemons are obscured, by addressing issues such as lost information from mask images and the inability to detect hidden tips.

4.4 comparative experiments with existing methods

In this chapter, we compare the existing harvesting method using a ring with the proposed method utilizing a system implemented on an

Android smartphone. The comparison focuses on the harvesting speed of lemons, as well as the usability and user experience of each method. Two experiments were conducted in this study: one in June 2024 using green lemons grown indoors (house cultivated) and another in October 2024 using green lemons grown outdoors. Both methods were evaluated to determine their practical feasibility and effectiveness in real-world scenarios. The experimental settings and results are described in the following sections.

4.4.1 Details of Subjects

To compare the existing harvesting method using a ring with the proposed system, experiments were conducted with the cooperation of students and staff from University of Yamanashi, employees from a company in Hiroshima Prefecture, and lemon farmers in Japan. For the experiment targeting indoor-grown lemons, 11 participants were involved, while 15 participants took part in the experiment targeting outdoor-grown lemons. As shown in Fig. 11, in experiment (a) with indoor-grown lemons, all participants were either beginners or had only a few instances of lemon harvesting experience. On the other hand, in experiment (b) with outdoor-grown lemons, one cooperating lemon farmer had 8 years of experience, and one participant working in agriculture-related employment had harvested lemons 10 times. Apart from 2 experiments, all other participants were either beginners or had only a few instances of lemon harvesting experience. It noted that the participants with 1–2 instances of harvesting experience in the outdoor experiment were the same individuals who participated in the greenhouse experiment.

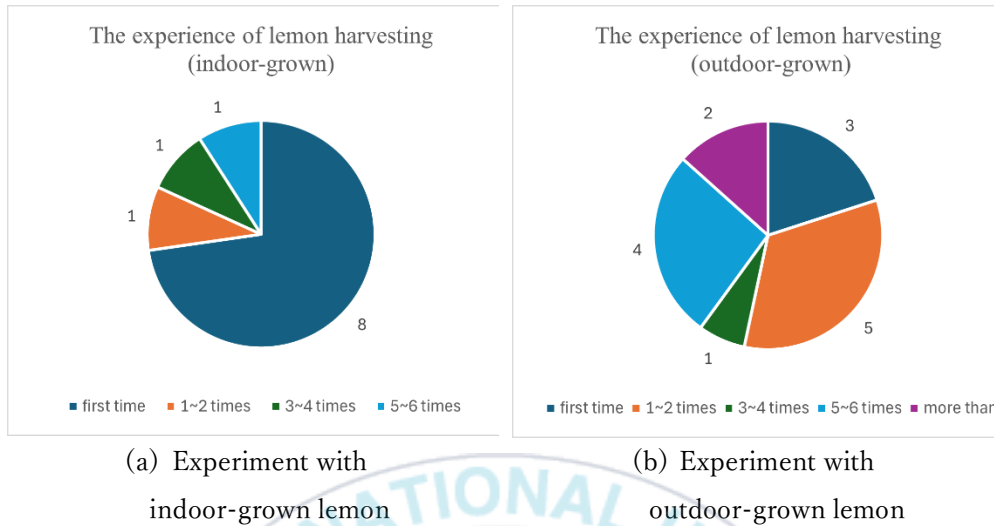


Fig. 11. Experience in lemon harvesting of subjects

4.4.2 Experimental Flow

The 2 experiments differed slightly in their content. In the June experiment, participants harvested only lemons located at low positions that did not require the use of a stepladder. In contrast, the October experiment included harvesting both low-position lemons and high-position lemons, which required the use of a stepladder. The experimental procedure is illustrated in Fig. 12.

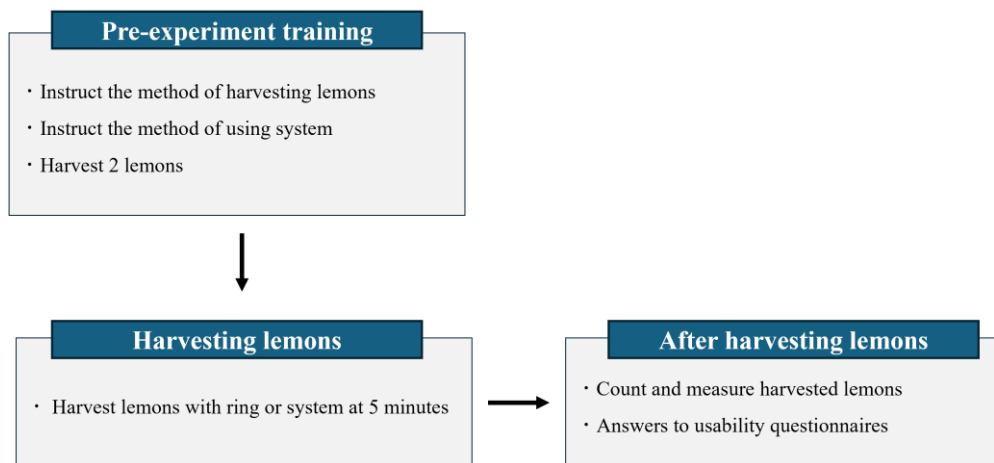


Fig. 12. The flow of comparative experiment

From Fig. X, during the pre-experiment practice, participants received instructions on harvesting methods and tool usage from a lemon farmer, as well as guidance on how to use the system from the system developers. To familiarize participants with limited lemon harvesting experience, they harvested 2 lemons before the actual experiment. For the harvesting task, the location of the lemons to be harvested was predetermined for each trial. Participants conducted the harvesting task using either the ring or the system within a 5-minute time limit. After each harvesting session, the number and size of the harvested lemons were recorded, and participants completed a questionnaire for usability and user experience. In the June experiment, each participant performed 2 harvesting trials: "ring-low" and "system-low" followed by a questionnaire and measurements. In the October experiment, each participant completed 4 harvesting trials: "ring-low", "ring-high", "system-low" and "system-high" with corresponding questionnaires and measurements.

4.4.3 Experimental Setup

The system was implemented with an RGB-D camera, an Android smartphone, and an edge server. The overall implementation is shown in Fig. 13. As shown in Fig. 13, first, an RGB-D camera connected to the smartphone captures RGB-D images. Next, the depth images obtained on the smartphone apply post-processing, such as hole filling and conversion to color images. These processed RGB-D images are then transmitted to the edge server. Fig. 14 illustrates the bounding boxes displayed on the smartphone for the detected lemons, with the colors of the boxes indicating different meanings based on the

estimation results. A red bounding box signifies that the estimated diameter meets or exceeds the harvest standard size, indicating the lemon is ready for harvest (Fig. 14 (a)). A blue bounding box signifies that the estimated diameter is below the harvest standard size, indicating the lemon should not be harvested (Fig. 14 (b)). A white bounding box indicates that the lemon is outside the optimal distance range due to the limitations of the RGB-D camera, making measurement impossible (Fig. 14 (c)). Using this result image, harvesters can easily determine which lemons to pick based on the information displayed on the smartphone.

In this study, the RGB-D camera used for the June experiment was the RealSense D435f, while for the October experiment, both the RealSense D435if and RealSense D435 were used. The Android smartphone utilized was the Google Pixel 7 Pro, and the edge server was a Jetson Orin Nano. Lemon and tip detection were performed using the YOLOv8n model, and regression calculations were executed using the Random Forest algorithm.

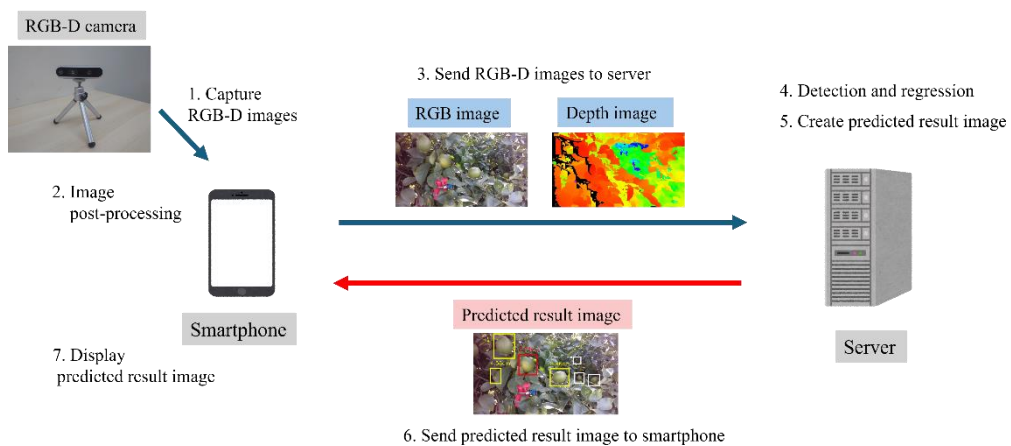


Fig. 13. The flow of proposed system with smartphone and edge server

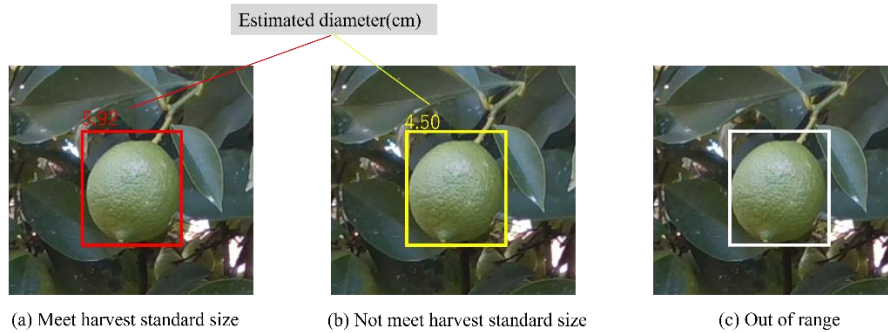


Fig. 14. Meaning of the frame and color of the estimated image

4.4.4 Experimental results

This section presents the results of the two experiments. Tables 8, 9, and 10 show the number of lemons harvested within 5 minutes and accuracy. The accuracy is defined as the proportion of harvested lemons meeting the required size criteria, for indoor low positions, outdoor low positions, and outdoor high positions, respectively.

	Average number of harvested	Minimum	Maximum	Standard deviation	Accuracy (%)
Ring	20.2	14	18	5.50	97.29
System	9.1	7	12	3.81	93.03

Table 8. The experiment results in indoor-grown lemons at low positions

	Average number of harvested	Minimum	Maximum	Standard deviation	Accuracy (%)
Ring	20.5	16	37	5.50	98.75
System	8.3	2	17	3.81	76.53

Table 9. The experiment results in outdoor-grown lemons at low positions

	Average number of harvested	Minimum	Maximum	Standard deviation	Accuracy (%)
Ring	13.6	5	24	4.98	95.30
System	7.3	4	16	2.87	91.45

Table 10. The experiment results in outdoor-grown lemons at high positions

In the experiment with indoor-grown lemons, participants using the ring harvested an average of 20.2 lemons, while those using the system harvested only 9.1 lemons, less than half. Similarly, in the experiment with outdoor-grown lemons, the number of lemons harvested decreased from an average of 20.5 with the ring to 8.3 with the system, showing a larger gap than in the indoor experiment. According to Table 10, in the experiment under the high-position condition with outdoor-grown lemons, participants harvested an average of 13.6 lemons using the ring and 7.3 lemons using the system. Comparing the results for low and high positions, the number of lemons harvested with the ring decreased significantly from about 20 lemons at low positions to 13.6 lemons at high positions. In contrast, the system showed a smaller decrease, with the number of lemons harvested dropping from 8.3 at low positions to 7.3 at high positions, resulting in a smaller gap compared to the ring. Additionally, when using the ring, the proportion of harvested lemons meeting the size criteria exceeded 95% in all experiments, demonstrating consistently high accuracy. When using the system, the accuracy was above 90% in all conditions except for outdoor-grown lemons at low positions, where the accuracy dropped significantly. Under the same low-position conditions, the accuracy decreased from 93.03% for indoor-grown lemons to 76.53% for outdoor-grown lemons,

a reduction of 16.5%. Similarly, under the same outdoor-grown conditions, the accuracy for high positions was 91.45%, but it decreased by 14.92% for low positions. This decline in accuracy is believed to be related to the amount of foliage and the density of lemons. Outdoor-grown lemons have significantly more leaves compared to indoor-grown lemons, and even lemons on the outer parts of the tree are often partially hidden by leaves. As a result, the accuracy of diameter estimation may have decreased due to occlusion. Additionally, for outdoor-grown lemons, those at low positions are typically more numerous and more densely clustered in the same area compared to those at high positions. These factors are considered to have contributed to the lower accuracy. Throughout the experiments, the number of lemons harvested using the system was consistently lower than when using the ring.

Possible reasons for this include insufficient practice with the system, the processing time required for each image, the difficulty of matching images to the actual lemons, and the limitations of the depth camera in outdoor conditions. The proposed system was hypothesized to offer the advantage of measuring the sizes of multiple lemons simultaneously compared to conventional methods. However, during observations, some participants captured multiple lemons from a distance, while others focused on one target at a time, inspecting each lemon individually. We suggest that additional training sessions with a few trials are necessary for effective use. Furthermore, the system demonstrated instability when used in real-time. Analysis of the images stored on the server with harvest-determination overlays revealed that

some lemons were displayed as harvestable in one frame but lost their bounding box in the next frame. This inconsistency made it difficult to match the same lemons across frames, further complicating the harvesting process. In addition to the system's unstable working, another issue identified was the difficulty in matching the lemons displayed on the screen with the lemons the human saw. This problem is thought to have 2 primary causes. The first cause arises when lemons are densely clustered, making it challenging to correspond the lemons displayed on the screen with the actual ones. An example of this scenario is shown in Fig. 15.

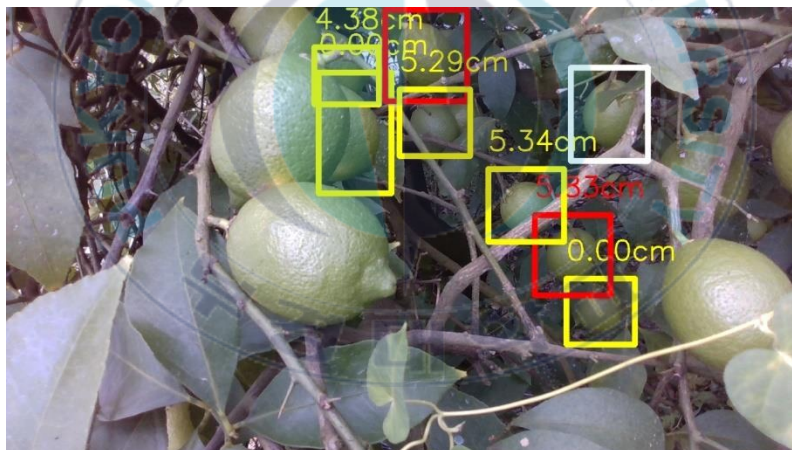


Fig. 15. Example of difficulty in mapping lemons to each other

From Fig. 15, it can be observed that some lemons are located close to each other in the image. It causes the displayed bounding boxes to overlap. As a result, when a user looks back and forth between the lemons and the image to identify the lemon within the red box, the image may update before they can recognize the target lemon. This makes it more challenging to identify the correct lemon. The second cause is the delay between capturing an image and displaying it on the

screen, despite real-time image updates. Fig. 16 shows the processing time taken from capturing to displaying an image for indoor-grown lemons in June and outdoor-grown lemons in October.

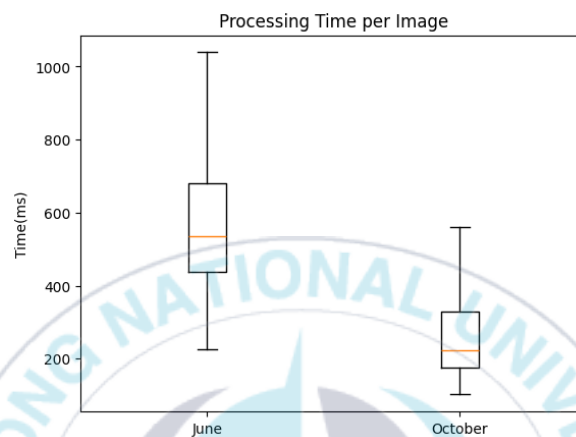


Fig. 16. The processing time per image

From Fig. 16, the average processing time per image was approximately 550ms in the June experiment, while in October, improvements to the network connection and server-side code reduced this to about 230ms per image. When the processing time exceeded the average, it was likely due to a large number of lemons in a single image, requiring additional time for detecting each lemon's tip and performing regression calculations. This delay creates a problem where even slight movements by the worker before the image update makes it more difficult to match the lemons on the screen with those in reality. Therefore, achieving shorter processing times is a challenge task for future development.

Another factor to consider is the instability of the depth camera's accuracy. In this experiment, we used three cameras from the RealSense D400 series: D435f, D435if, and D435. The D435if model includes an

inertial measurement unit (IMU) but otherwise shares the same architecture as the D435f. Both the D435f and D435if are equipped with a short-range infrared filter, which was expected to make them suitable for outdoor use under direct sunlight. In the experiment with indoor-grown lemons, the D435f camera seemed to work correctly compared with the outdoor situation. However, for outdoor use, it was observed that the cameras equipped with infrared filters, such as the D435f and D435if, failed to capture accurate depth information at specific distances, approximately 40 cm to 50 cm. In contrast, the D435, which lacks an infrared filter, was found to provide more accurate depth information. As a result, it is evident that the current limitations of RGB-D cameras, particularly under outdoor conditions, pose challenges for this system. Further exploration of hardware adjustments or alternative devices is required for future improvements.

Considering these factors, future challenges include the integration of Augmented Reality (AR) technology with smart glasses to overlay harvestable lemons, achieving faster processing times, and selecting depth cameras better suited for outdoor activities. As a result, these improvements are expected to enhance the system's usability and reliability in practical applications.

4.4.5 Subjective Evaluation for Usability

In this experiment, participants were asked to complete a usability and user experiment questionnaire after each lemon harvesting task. We used User Experience Questionnaire (UEQ) [19], which is a standardized tool designed to evaluate the usability and user experience of products, services, or systems. This method consists of 26 pairs of

contrasting impression words such as "fast/slow" or "annoying/enjoyable". It evaluates 7 scales. The examples of these word pairs are shown in Fig. 17.

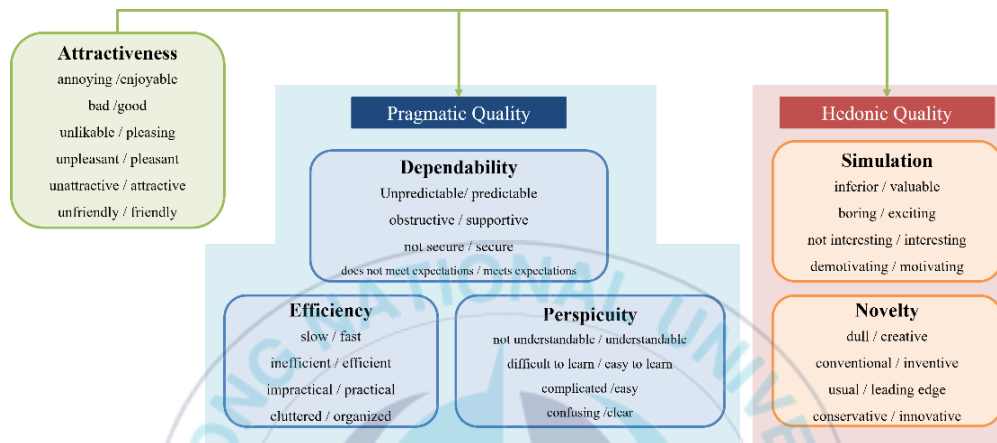


Fig. 17. UEQ word impression word pairs

As shown in Fig. 17, the impression word pairs are broadly categorized into three main dimensions: "Attractiveness," "Pragmatic Quality," and "Hedonic Quality." Furthermore, "Pragmatic Quality" is subdivided into dependability efficiency, and, perspicuity while "Hedonic Quality" is divided into stimulation and novelty. Table 11 presents the results of Attractiveness, Pragmatic Quality, and Hedonic Quality for lemon harvesting under 3 conditions: indoor-grown lemons at low positions, outdoor-grown lemons at low positions, and outdoor-grown lemons at high positions. The values in Table 11 represent the average scores for each category, rated on a 7 scale ranging from -3 to 3. Lower values indicate a negative impression, while higher values represent a positive impression. A larger value is an indicator of a superior method, suggesting that the approach performs more

effectively or delivers better outcomes when compared to other methods.

	Place	Position	Attractiveness	Pragmatic Quality	Hedonic Quality
Ring	Indoor	Low	1.00	1.73	-0.57
	Outdoor	Low	0.72	1.63	-0.56
	Outdoor	High	0.21	0.71	-0.55
Average of ring			0.64	1.36	-0.56
System	Indoor	Low	1.48	0.47	1.84
	Outdoor	Low	0.63	-0.26	1.31
	Outdoor	High	0.38	-0.37	1.13
Average of system			0.83	-0.05	1.43

Table 11. The scales for attractiveness, pragmatic quality, and hedonic quality under 3 conditions

For Attractiveness, significant differences were observed between indoor and outdoor conditions. In the case of indoor situations, the scores for the ring and system were 1.00 and 1.48, respectively. In outdoor conditions, the scores decreased to 0.72 for the ring and 0.63 for the system at low positions. Furthermore, for outdoor at high positions, the scores dropped to less than half of those for outdoor at low positions, highlighting a notable decline. For Pragmatic Quality and Hedonic Quality, distinct trends were observed between the ring and the system. In terms of Pragmatic Quality, the ring scored 1.36, while the system scored -0.05. Ring indicates that practical aspects such as ease of use, efficiency, and safety were rated higher for the ring. In contrast, for Hedonic Quality, the system significantly outperformed the ring, with an average score of 1.43 compared to the ring's average

of -0.56. This suggests that the system provided a much more positive impression in terms of stimulation and novelty. The most notable differences between the ring and the system were in Efficiency and Novelty. These results are illustrated in Fig. 18 and Fig. 19, respectively.

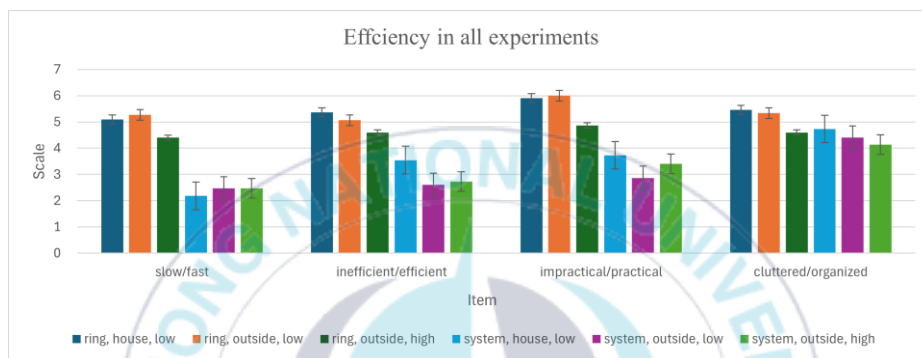


Fig. 18. The results of efficiency items for all experiments

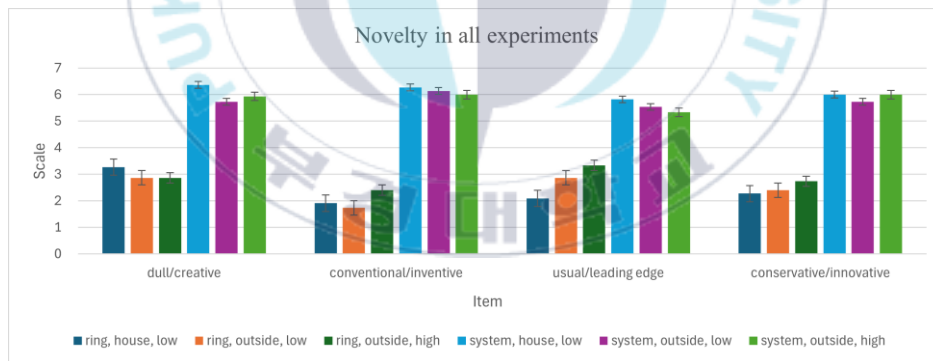


Fig. 19. The results of novelty items for all experiments

For Efficiency, the "slow/fast" metric showed the largest difference between the ring and the system. In all conditions, the ring consistently scored above 4, while the system scored below 3 in every case, indicating a clear advantage for the ring in terms of speed and efficiency. For Novelty, the "conventional/inventive" metric exhibited the most significant difference. The ring received scores around 2, reflecting a more conventional impression, whereas the system scored

near 6, demonstrating a very high score and a strong perception of innovation.

From the results of the UEQ questionnaire, the system received higher scores than the ring for aspects related to enjoyment and visual appeal. However, for practical usability and speed, the system was rated lower than the ring. The ring's straightforward and intuitive method of passing it through the lemon likely contributed to its positive evaluation. In contrast, the system's lower scores can be attributed to the disadvantages outlined in section 4.3.3, such as slower processing and difficulties in use.

The high scores for the system may stem from its novelty, as it utilizes a smartphone and camera, offering a distinctly modern and innovative approach compared to existing harvesting methods. To make the system suitable for practical use in the field, it is necessary not only to improve measurable aspects such as estimation accuracy and image processing time but also to enhance the user interface (UI). Designing a clearer and more intuitive display method could significantly improve its usability and acceptance.

V. CONCLUSION

We propose a method to estimate the diameter of lemons using RGB-D images. Through our experiments, the accuracy of lemon and tip detection, as well as size estimation, was verified. Using YOLOv8n enabled fast and accurate detection of lemons and tips. By incorporating five features, including the tip position of the lemon from the detection results and depth information, into a regression model, we confirmed that the method is independent of the lemon's orientation. These results indicate that the proposed method is practical for supporting lemon harvesting operations in actual orchards. Moreover, the versatility of our method suggests that it can be applied to other fruits and vegetables that require precise size measurement, considering their orientation. This makes our approach broadly applicable in agricultural practices where non-contact and accurate size estimation is crucial. Ultimately, this approach has the potential to be integrated into automated harvesting systems for various crops, reducing labor costs and improving the quality and efficiency of harvesting operations across different agricultural contexts. This advancement supports the broader goal of developing smart agriculture solutions that leverage cutting-edge technology to enhance productivity and sustainability. Future work should focus on improving accuracy in scenarios where the targets are partially obscured by leaves or other objects. Expanding the dataset to include more diverse conditions and refining the feature extraction process could further enhance the robustness and reliability of the method.

References

- [1] P. Buayai, K. R. Saikaew, and X. Mao, "End-to-End Automatic Berry Counting for Table Grape Thinning," *IEEE Access*, vol. 9, pp. 4829–4842, 2021, doi: 10.1109/ACCESS.2020.3048374.
- [2] Xingxu Li, Nan Ma, Yiheng Han, Shun Yang, and Siyi Zheng, "AHPPEBot: Autonomous Robot for Tomato Harvesting based on Phenotyping and Pose Estimation," *IEEE International Conference on Robotics and Automation (ICRA)*, 2024, Yokohama, Japan.
- [3] Junxiong Liang, Kai Huang, Huan Lei, Zhenyu Zhong, Yingjie Cai, and Zeyu Jiao, "Occlusion-aware fruit segmentation in complex natural environments under shape prior," *Computers and Electronics in Agriculture*, vol. 217, article no. 108620, Feb. 2024, doi: 10.1016/j.compag.2024.108620.
- [4] Zania S Pothan and Stephen Nuske, "Texture-based Fruit Detection via Images using the Smooth Patterns on the Fruit," *IEEE International Conference on Robotics and Automation (ICRA)*, 2016, Stockholm, Sweden, doi: 10.1109/ICRA.2016.7487722.
- [5] Dandan Wang and Dongjian He, "Fusion of Mask RCNN and attention mechanism for instance segmentation of apples under complex background," *Computers and Electronics in Agriculture*, vol. 196, article no. 106864, May 2022, doi: 10.1016/j.compag.2022.106864.
- [6] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick, "Mask R-CNN," *IEEE International Conference on Computer Vision (ICCV)*, 2017, Venice, Italy, doi: 10.1109/ICCV.2017.322.

- [7] Daniel Bolya, Chong Zhou, Fanyi Xiao, and Yong Jae Lee, "YOLACT: Real-time Instance Segmentation," in the IEEE/CVF international conference on computer vision, 2019, pp. 9157–9166.
- [8] Jocher, G., Chaurasia, A. and Qiu, J., 2023. Ultralytics YOLOv8. [Online]. Available: <https://github.com/ultralytics/ultralytics>.
- [9] Jocher, G and Qiu, J, 2024. Ultralytics YOLOv11. [Online]. Available: <https://github.com/ultralytics/ultralytics>.
- [10] R. Sapkota, D. Ahmed, M. Churuvija and M. Karkee, "Immature Green Apple Detection and Sizing in Commercial Orchards Using YOLOv8 and Shape Fitting Techniques," in IEEE Access, vol. 12, pp. 43436-43452, 2024, doi: 10.1109/ACCESS.2024.3378261.
- [11] Zhenglin Wang, Kerry B. Walsh, and Brijesh Verma, "On-Tree Mango Fruit Size Estimation Using RGB-D Images," Sensors, vol. 17(12), article no. 2738, 2017, doi: 10.3390/s17122738.
- [12] Otsu, N., "A threshold selection method from gray-level histograms", IEEE Trans. Syst. Man Cybern. 9 (1), 62–66, 1979.
- [13] Anders Grunnet-Jepsen and Dave Tong, "Depth Post-Processing for Intel® RealSense™ D400 Depth Cameras," Intel, [Online]. Available: <https://www.intel.com/content/dam/support/us/en/documents/emerging-technologies/intel-realsense-technology/Intel-RealSense-Depth-PostProcess.pdf>. [Accessed: July 26, 2024].
- [14] Robert Tibshirani, "Regression Shrinkage and Selection via the Lasso," Journal of the Royal Statistical Society. Series B (Methodological), Vol. 58(1), pp. 267-288, 1996.

- [15] E. Hoerl and Robert W. Kennard, "Ridge Regression: Biased Estimation for Nonorthogonal Problems," *Technometrics*, vol.12(1), pp.55-67, 1970.
- [16] Hui Zou and Trevor Hastie, "Regularization and variable selection via the elastic net," *Journal of the Royal Statistical Society. Series B (Statistical Methodology)*, vol. 67(2), pp. 301-320 , 2005.
- [17] Leo Breiman, "Random Forests", *Machine Learning*, vol.45, pp. 5–32, 2001.
- [18] Zhaowei Cai and Nuno Vasconcelos, " Cascade R-CNN: High Quality Object Detection and Instance Segmentation", *Conference on Computer Vision and Pattern Recognition(CVPR)*, 2018, pp. 6154-6162
- [19] Laugwitz, B., Schrepp, M. & Held, T. (2008). Construction and evaluation of a user experience questionnaire. In: Holzinger, A. (Ed.): *USAB 2008, LNCS 5298*, pp. 63-76.

List of Publications

Conference

1. Ayuna Dohi, Xiaoyang Mao, Prawit Buayai, Ki-Ryong Kwon, "The Estimation of Lemon Size Using RGB-D Camera and Deep Learning," in KMMS Conference 2023, Nov. 2023.
2. Ayuna Dohi, Buayai Prawit, Ki-Ryong Kwon and Xiaoyang Mao, "Tilt-Invariant Lemon Size Estimation Using RGB-D Camera Images" in Cyberworlds2024, Oct. 2024



Acknowledgment

I researched my study in PKNU as a part of Real Problem Solving Driven Artificial Intelligence Education Program(A3I) was agreed upon by 4 universities: University of Yamanashi in Japan, Hangzhou Dianzi University in China, Pukyong National University in South Korea and University of Malaysia Perlis in Malaysia.

Firstly, I would like to thank my supervisor, Prof. Ki-Ryong Kwon, for his constant guidance and for providing the necessary equipment, information, and direction to accomplish the present work.

Secondly, I would like to thank my supervisor in Japan, Prof. Xiaoyang Mao, and assistant Prof. Prawit Buayai. I received invaluable guidance on my research, including insightful feedback on the content and clear directions for conducting experiments.

Thirdly, This study was conducted as part of the Hiroshima Smart Agriculture Promotion Project. I would like to extend my heartfelt gratitude to Enecom, Inc. in Hiroshima Prefecture for consistently organizing data collection and coordinating the experiment schedules, as well as to the companies and farmers in Hiroshima Prefecture who kindly offered their cooperation.

Finally, I would like to express my gratitude to all the members of the research laboratories in both South Korea and Japan.