Article

Adaptive Active Positioning of *Camellia oleifera* Fruit Picking Points: Classical Image Processing and YOLOv7 Fusion Algorithm

Yunhe Zhou [1,2], **Yunchao Tang** [1,*], **Xiangjun Zou** [2], **Mingliang Wu** [3,*], **Wei Tang** [2], **Fan Meng** [1], **Yunqi Zhang** [1] **and Hanwen Kang** [4]

1 College of Urban and Rural Construction, Zhongkai University of Agriculture and Engineering, Guangzhou 510225, China
2 Foshan-Zhongke Innovation Research Institute of Intelligent Agriculture and Robotics, Foshan 528000, China
3 College of Mechanical and Electrical Engineering, Hunan Agricultural University, Changsha 410128, China
4 College of Engineering, South China Agricultural University, Guangzhou 510642, China
* Correspondence: ryan.twain@zhku.edu.cn (Y.T.); mlwu@hunau.edu.cn (M.W.);
  Tel.: +86-135-7025-7183 (Y.T.); +86-139-7492-3150 (M.W.)

Abstract: *Camellia oleifera* fruits are randomly distributed in an orchard, and the fruits are easily blocked or covered by leaves. In addition, the colors of leaves and fruits are alike, and flowers and fruits grow at the same time, presenting many ambiguities. The large shock force will cause flowers to fall and affect the yield. As a result, accurate positioning becomes a difficult problem for robot picking. Therefore, studying target recognition and localization of *Camellia oleifera* fruits in complex environments has many difficulties. In this paper, a fusion method of deep learning based on visual perception and image processing is proposed to adaptively and actively locate fruit recognition and picking points for *Camellia oleifera* fruits. First, to adapt to the target classification and recognition of complex scenes in the field, the parameters of the You Only Live Once v7 (YOLOv7) model were optimized and selected to achieve *Camellia oleifera* fruits' detection and determine the center point of the fruit recognition frame. Then, image processing and a geometric algorithm are used to process the image, segment, and determine the morphology of the fruit, extract the centroid of the outline of *Camellia oleifera* fruit, and then analyze the position deviation of its centroid point and the center point in the YOLO recognition frame. The frontlighting, backlight, partial occlusion, and other test conditions for the perceptual recognition processing were validated with several experiments. The results demonstrate that the *precision* of YOLOv7 is close to that of YOLOv5s, and the *mean average precision* of YOLOv7 is higher than that of YOLOv5s. For some occluded *Camellia oleifera* fruits, the YOLOv7 algorithm is better than the YOLOv5s algorithm, which improves the detection accuracy of *Camellia oleifera* fruits. The contour of *Camellia oleifera* fruits can be extracted entirely via image processing. The average position deviation between the centroid point of the image extraction and the center point of the YOLO recognition frame is 2.86 pixels; thus, the center point of the YOLO recognition frame is approximately considered to be consistent with the centroid point of the image extraction.

Keywords: machine vision; YOLOv7; image processing; grabbing point positioning

1. Introduction

Camellia oleifera C. Abel is a kind of subtropical evergreen shrub, which is widely planted as an oil plant in many countries [1]. In addition, tea oil has high unsaturated fatty acid content and elevated nutritional value [2]. In the process of *Camellia oleifera* fruits' harvest, a large amount of labor is needed, and the efficiency of picking fruit by labor is

not high and the labor cost is high. Therefore, fruit farmers also cultivated dwarf varieties and fruit larger oil tea trees, with intelligent mechanized picking instead of the traditional manual picking. Because the flower of *Camellia oleifera* C. Abel grows at the same time as the fruit, the large shock force will cause the flower to fall and affect the yield. At the same time, considering that the *Camellia oleifera* fruit is a spherical fruit, only the claw type picking execution end can be used. Considering the particularity of fruit growth, when using the claw, there may be an incorrect selection of grabbing points, which will reduce the success rate of picking. Therefore, precise positioning has become a difficult problem for robot operation.

Camellia oleifera fruits grow randomly and are easily covered by leaves. The color of leaves and fruits is similar, or there is overlap or occlusion between fruits. At the same time, different lighting conditions also have a great impact on the identification and positioning of *Camellia oleifera* fruits. Therefore, many scholars have studied the recognition and location of fruit. Yu et al. used RGB-D camera to collect the color and depth images of lychee, eliminating redundant image information by depth image segmentation, and identifying lychee fruits by a random forest binary classification model based on color and texture features [3]. Kang et al. developed and evaluated a visual method of autonomous apple harvesting based on full depth learning. This method realizes fruit recognition, detection and estimation of the correct method attitude of each fruit, which is used for the robot arm to perform picking. The harvesting success rate of the robot harvesting system is 0.8, and the cycle time is 6.5 s [4]. Li et al. used an RGB depth (RGB-D) camera and semantic segmentation method to study the randomly distributed lychee clusters, and obtained a lychee fruit with a detection accuracy of 83.33% and location accuracy of $17.29^{\circ} \pm 24.57^{\circ}$ [5]. Chen et al. created a measurement framework based on multi-vision technology for bananas, providing a practical and theoretical reference for the three-dimensional sensing of banana central stocks in complicated environments [6]. Wu et al. used a different method based on the fusion of deep learning and edge detection to classify and recognize bananas, inflorescence axes and flower buds, and automatically located the cutting point on the inflorescence axis [7]. Lin et al. proposed a probabilistic image segmentation method and developed a global point cloud descriptor based on angle/color/shape to collect the feature vector of the entire point cloud. The detection *precision* was 0.864,0.886 and 0.888, respectively [8]. Santos et al. used CNNs to detect, decollate and track grape clusters with complex colors, shapes, compactness, and sizes. The test set contains 408 grape cluster images of a trellis-system based vineyard, and the *F1* score was 0.91 [9]. Xu et al. took the ripe *Camellia oleifera* fruits as the research object, used the point cloud data to discriminate the *Camellia oleifera* fruits, and determined the position and direction angle of the *Camellia oleifera* fruits' picking center point in the three-dimensional space [10]. Gao et al. proposed a multi-class apple detection means for fruit trees with dense leaves based on the Faster Region-Convolutional Neural Network. It detected apples under different terms of fruit-occluded, wire/branch-occluded, leaf occluded and non-occluded fruit, with an *average precision* of 0.848, 0.858, 0.899, and 0.909, respectively [11]. Tang et al. described the difficulties of fruit recognition and location: target identification in the scene of occlusion and light changes environments, target tracking in dynamic jamming environments, 3D target reconstruction, and fault tolerance of the vision system for agricultural robots [12]. The strawberry robot designed by Xiong et al.,was equipped with internal sensors where the gripper could sense and correct the position error; it had a robustness to positioning error introduced by the vision module. The robot can select isolated strawberries with a near perfect success rate (96.8%) [13].

Some scholars use the YOLO algorithm to detect and identify fruits. Zhang et al. proposed a panoramic method of viewing the full fruit yield and counting based on deep learning object detection; the researchers verified this via holly trees with dense fruits. Accurate fruit counting has good robustness to shadows, coverage, and incomplete contours [14]. Ji et al. raised an apple target detection method based on Shufflflenetv2-YOLOX. The *precision*, *recall*, *average precision*, and *F1* of the trained network under the

verification set are 95.62%, 93.75%, 96.76%, and 0.95, respectively, and the detection speed reaches 65 frames per second [15]. Wang et al. proposed the DSE-YOLO to detect multi-stage strawberries, with the *mean average precision* value of 86.58% and *F1* score value of 81.59% [16]. Tian et al. raised an improved YOLO-V3 model to detect apples at different growth stages in gardens with a fluctuating illumination, sophisticated background, and overlapping apples, leaves, and branches. The results demonstrated that the model has an average detection time of 0.304 s per frame at a resolution of 3000 × 3000 [17]. Koirala et al. compared the performance of Faster R-CNN (VGG) and Faster R-CNN (ZF) and single-stage technologies of YOLOv2, YOLOv2 (tiny), YOLOv3 and SSD in detecting mango fruits in tree crown images. Based on YOLOv2 (tiny) and YOLOv3, a new architecture "MangoYOLO" has also been developed [18]. Gao et al. established an automatic counting method based on deep learning and trunk tracking. The *mean average precision (mAP)* of the trunk and fruit detection was 99.35%. In an orchard video, the counting accuracy of the manual counting method was 91.49%, and the correlation coefficient R^2 was 0.9875, with the counting performed by manual counting [19]. Parico et al. used the object detection model YOLOv4 and the multi object tracking algorithm depth SORT mobile application to generate a powerful real-time pear fruit counter [20]. Mirhaji et al. used the YOLO detection model to detect and count oranges. The *mean average precision (mAP)*, *precision*, *recall*, and *F1* score of YOLOV4 as the best model for orange detection were 90.8%, 91.23%, 92.8%, and 92%, respectively [21]. MacEachern et al. developed a deep learning artificial neural network model of wild blueberry at maturity, and the model was used for yield estimation [22]. Tang et al. improved the YOLOv4 tiny model and detected the fruits under various lighting conditions. Although the lighting changes were significant, the algorithm still shows a high positioning stability and robustness [23]. Lv et al. used the improved YOLOv5 depth learning algorithm to put forward a visual recognition method for apple growth morphology in the orchard. The *mean average precision (mAP)* reached 98.4%, and the *F1* score was 0.928 [24]. Wang et al. proposed a precise apple fruit detection method with a small model size based on the channel pruning YOLOv5s algorithm. The *recall* rate, *precision*, *F1*-score and false detection rate were 87.6%, 95.8%, 91.5%, and 4.2%, respectively [25]. Yan et al. proposed a lightweight apple target detection method for picking robots using the improved YOLOv5s. The improved network model can effectively identify graspable apples that are not occluded or only occluded by leaves and non-graspable apples that are occluded by branches or other fruits. Among them, the identification *recall*, *precision*, *mAP*, and *F1* are 91.48%, 83.83%, 86.75%, and 87.49%, respectively [26]. Wu et al. studied the improved YOLOv5-B model for the banana multi-target recognition, and the (*mAP*) was 93.2%. The accurate relationship model of the geometric measurement between laser measurement and depth camera was established, and the positioning error was analyzed. Based on the YOLO algorithm, it provides convenience for fruit detection, localization, and recognition [27].

In this paper, the algorithm and image processing method are used to detect and locate *Camellia oleifera* fruits in a complex environment combined with visual perception, deep learning, and classical image processing and obtain the optimal parameters through continuous optimization selection. The main contributions of this paper are summarized as follows:

(1) Deep learning based on visual perception is used to detect *Camellia oleifera* fruits in intricate environments, and the *precision (P)*, *recall (R)*, and *mean average precision (mAP)* of the detection of YOLOv7 and YOLOv5s algorithms are compared.

(2) Through classical image processing, image preprocessing, segmentation, and morphological processing is carried out on the *Camellia oleifera* fruit image to find the centroid of the *Camellia oleifera* fruit; the position deviation between the centroid point of the *Camellia oleifera* fruit is obtained by image processing, and the center point in the YOLO recognition frame is analyzed.

The second section introduces the materials and data collection, the third section introduces the detection algorithm of the YOLOv7 *Camellia oleifera* fruit, the fourth section introduces the image processing methods, the fifth section introduces the test results and analysis, and the sixth section draws conclusions.

2. Material and Data Collection

2.1. Test Equipment

This experiment is based on the deep learning framework Python 1.8.0, using Python 3.8.5 as the programming language and Linux as the operating system with a ×86_ 64 processor, 300 G memory, and Tesla V100 video card. The acquisition devices are a mobile phone, camera, and computer (Figure 1), and the image resolution is 2400 × 1080 pixels. The software system is mainly based on the OpenCV function library and YOLOv7.
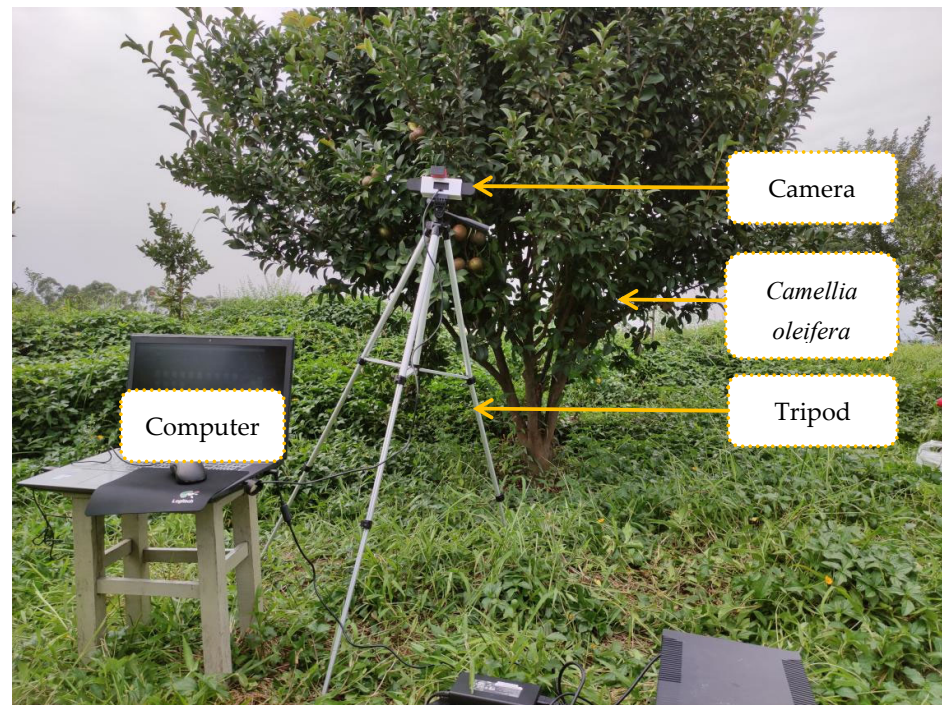

Figure 1. Experimental equipment.

2.2. Image and Data Collection

The *Camellia oleifera* fruits' data set used in this paper is sampled from the ancient tree garden of Meilin Lake Avenue, Huadu, Guangzhou, Guangdong, China. It is located at a longitude of 113.0449077° and the latitude of 23.4787190°. The *Camellia oleifera* fruit likes warmth and requires an annual average temperature of 14–21 °C. The light needs to be sufficient; the annual sunshine hours that the *Camellia oleifera* fruit used in this study are typically between 1800 and 2200 h, which is suitable for the growth of *Camellia oleifera* fruits. In this investigation, 1364 images of the *Camellia oleifera* fruit with smooth, backlight and partial occlusion are collected and used for training and testing (e.g., for the *Camellia oleifera* fruit detection algorithm) via data enhancement processing. The images of *Camellia oleifera* fruit with the frontlighting, backlight, and partial occlusion are shown in Figure 2. The collected images of the *Camellia oleifera* fruits are labeled with the label tool labelImg, that is, the selected *Camellia oleifera* fruits picked are framed; the data set includes 820 training sets, 272 validation sets, and 272 test sets.
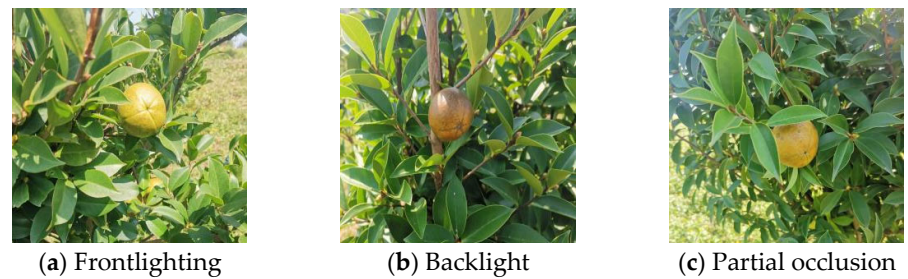
(**a**) Frontlighting   (**b**) Backlight   (**c**) Partial occlusion

Figure 2. *Camellia oleifera* fruit.

3. YOLOv7 *Camellia oleifera* Fruit Detection Algorithm

3.1. YOLOv7

YOLOv7 is a general depth learning model that cannot be directly applied to complex field scenes. Therefore, we optimize and select the model parameters to achieve the target detection of *Camellia oleifera* fruits. In this work, we apply the initial learning rate of 0.01, 0.005 and 0.0001. The loss function curve of the validation set obtained has an obvious trend of decline and then rise. After constant adjustment and optimization, the final initial learning rate is determined to be 0.0005. YOLOv7 was proposed by Alexey Bochkovskiy's team. In terms of architecture, it expands the aggregation network of the high efficiency layer and scales the model for concatenation-based models [28]. The network structure is shown in Figure 3.
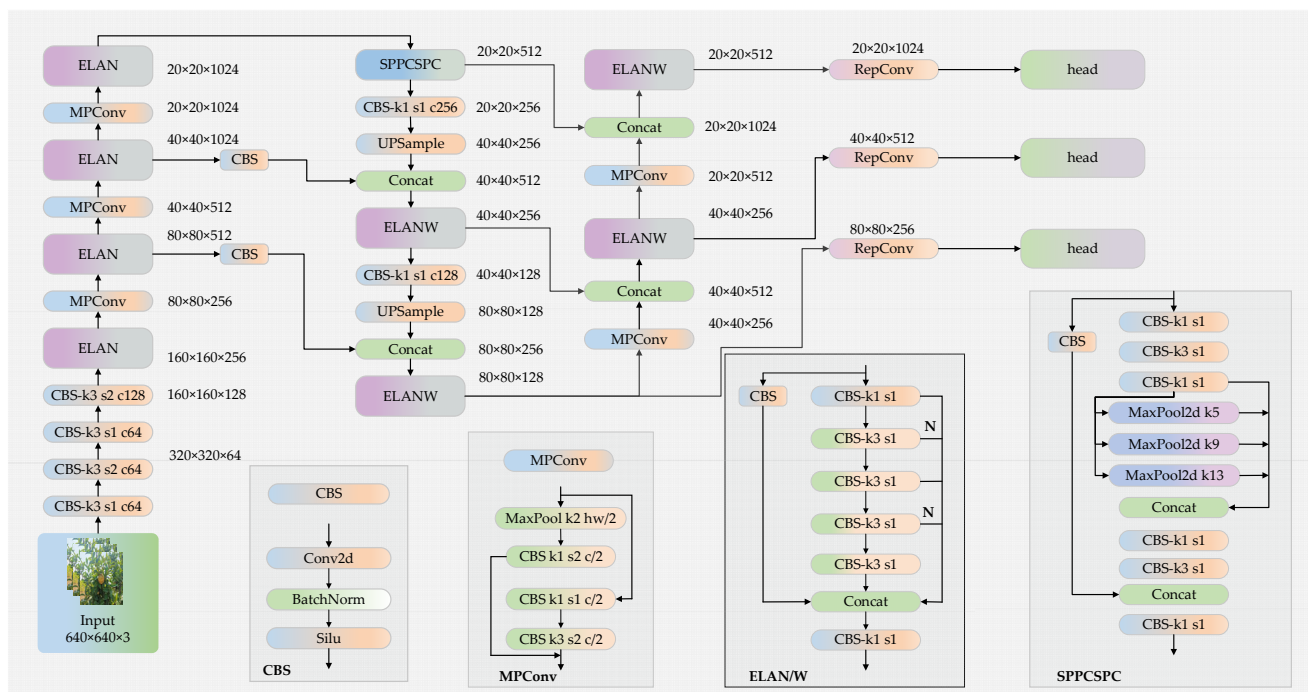

Figure 3. Network structure.

3.2. Model Training

During the algorithm training process, the Stochastic Gradient Descent (SGD) algorithm is used to optimize the algorithm. The momentum size of the momentum optimizer is set to 0.937, the attenuation coefficient is 0.0005, the number of target categories is 1, the number of training iterations is 200, and the number of samples' input for each iteration is 32. The algorithm with the best detection effect is selected as the *Camellia oleifera* fruit

detection algorithm. This experiment trains two algorithms in total, the YOLOv5s and YOLOv7 research experiments.

4. Image Processing

Image processing includes image preprocessing, image transformation, image enhancement and restoration, image segmentation, image feature extraction/selection, and image classification [29]. The specific practice flow chart of the *Camellia oleifera* fruit image in this paper is shown in Figure 4.
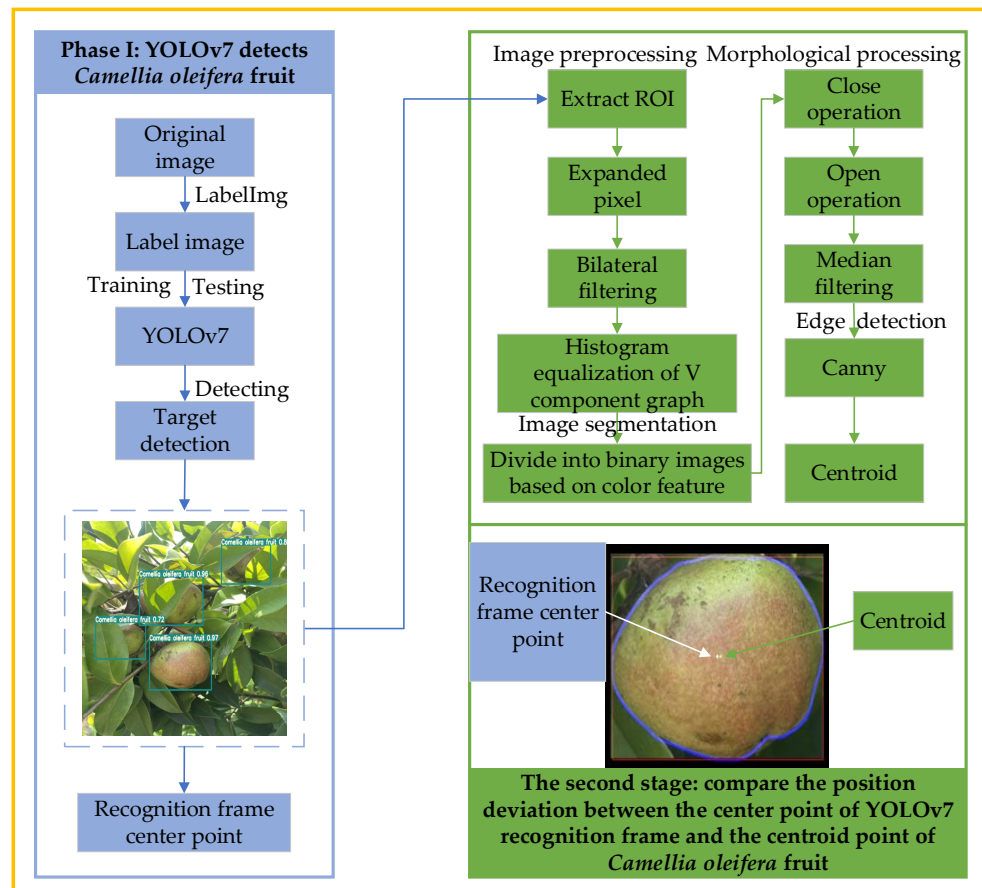

Figure 4. Flow chart.

4.1. Image Preprocessing

First, cut the YOLOv7 object detection picture to obtain the image of the region of interest (ROI) of the *Camellia oleifera* fruit. To avoid the interference of subsequent fruit stalks on the recognition area, we use the copyMakeBorder function command to copy the source image to the middle of the target image and form a border around the image; then, we expand the edge of the obtained *Camellia oleifera* fruit image by 10 pixels.

In image processing, it is normally necessary to reduce the impact of noise in the image through filtering. Common methods include mean filtering, Gaussian filtering, median filtering, and bilateral filtering. Bilateral filtering is a nonlinear filtering, which can achieve the noise removal and edge preservation effects [30]. Therefore, bilateral filtering of the pixel-expanded *Camellia oleifera* fruit image can reduce noise while maintaining the edge.

The *Camellia oleifera* fruits collected in this study are partially bright or dark due to uneven illumination, reflection, and other reasons, which is not conducive to the image

segmentation of *Camellia oleifera* fruits. The histogram equalization algorithm is used to equalize the light and dark degree of the *Camellia oleifera* fruit image. We transform the RGB image after bilateral filtering into the color space to obtain the HSV image. The V component in the three channel components of the converted HSV image is separated, and the V component is separately equalized via a histogram. The V component in the separated HSV three channel components is replaced by the image equalized by the V component histogram, and then the enhanced HSV image is generated by merging the components. Then, we convert the HSV image to an RGB image. The specific preprocessed image is shown in Figure 5.
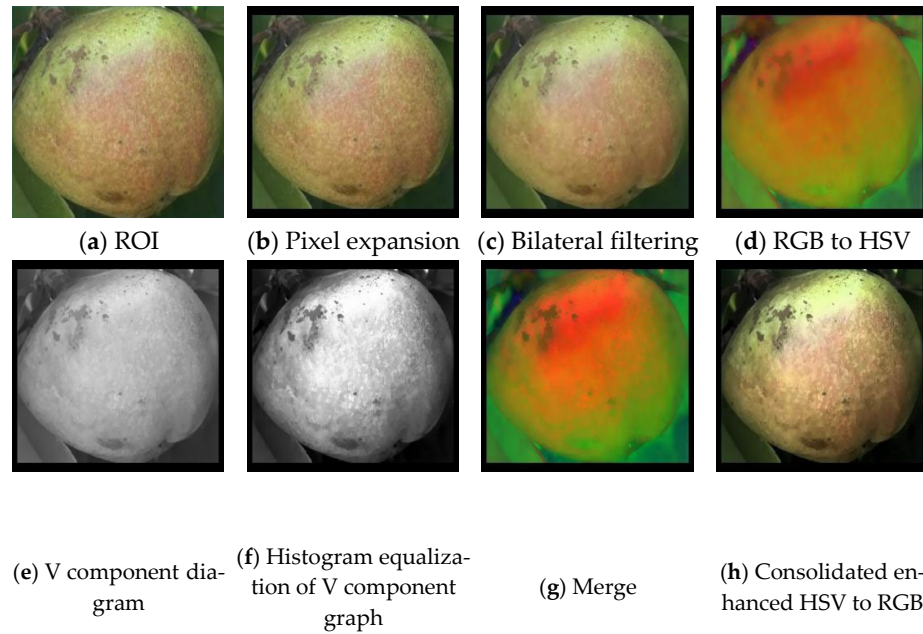
(**a**) ROI　　　(**b**) Pixel expansion　　(**c**) Bilateral filtering　　(**d**) RGB to HSV

(**e**) V component diagram　　(**f**) Histogram equalization of V component graph　　(**g**) Merge　　(**h**) Consolidated enhanced HSV to RGB

Figure 5. Image preprocessing.

The RGB image is a three-channel image, which is composed of red, green, and blue data of the same line and column; the HSV image also contains three channels of data of the same size: H channel, S channel and V channel [31]. The principle of RGB to HSV is shown in the following Equations (1)–(10) [32]. First, convert R, G, and B values to 0–1.

$$R = R/255 \tag{1}$$

$$G = G/255 \tag{2}$$

$$B = B/255 \tag{3}$$

$$V = \max(R, G, B) \tag{4}$$

$$S = \begin{cases} \dfrac{V - \min(R, G, B)}{V} & , \ if \ V \neq 0 \\ 0 & , \ else \end{cases} \tag{5}$$

$$H = \begin{cases} 60*(G-B)/(V-\min(R,G,B)) & , \ if \ V = R \\ 120 + 60*(B-R)/(V-\min(R,G,B)) & , if \ V = G \\ 240 + 60*(R-G)/(V-\min(R,G,B)) & , if \ V = B \end{cases} \tag{6}$$

If the calculated H is less than 0, add 360 to this value to obtain the final value:

$$H = H + 360 \tag{7}$$

Because OpenCV needs to visualize HSV images, the values need to be converted to 0–255, via:

$$H = H/2 \tag{8}$$

$$S = S*255 \tag{9}$$

$$V = V*255 \tag{10}$$

The principle of HSV to RGB is shown in the following Equations (11)–(19). First, convert the H, S, and V values of the visualization image to the range of 0–360, 0–1, and 0–1, respectively.

$$H = H*2 \tag{11}$$

$$S = S/255 \tag{12}$$

$$V = V/255 \tag{13}$$

Then, R, G, and B are calculated as follows, where the floor function represents a downward rounding operation:

$$h_i = floor(\frac{H}{60}) \tag{14}$$

$$f = \frac{H}{60} - h_i \tag{15}$$

$$p = V*(1-S) \tag{16}$$

$$q = V*(1-f*S) \tag{17}$$

$$t = V*(1-(1-f)*S) \tag{18}$$

$$(R,G,B) = \begin{cases} (V,t,p) & , \ if \ h_i = 0 \\ (q,V,p) & , \ if \ h_i = 1 \\ (p,V,t) & , \ if \ h_i = 2 \\ (p,q,V) & , \ if \ h_i = 3 \\ (t,p,V) & , \ if \ h_i = 4 \\ (V,p,q) & , \ if \ h_i = 5 \end{cases} \tag{19}$$
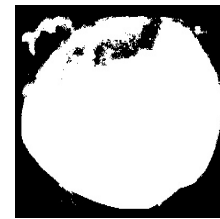
4.2. Image Segmentation

In this paper, HSV color space and gray scale factor are used to segment *Camellia oleifera* fruit, and the effects of the two methods are compared. Method 1: Segment *Camellia*

oleifera fruit based on HSV color space and segment the whole *Camellia oleifera* fruit based on color features through the inRange function to transform it into a binary image. Method 2: The RGB image is separated by a single channel to obtain a grayscale image. According to Equation (20), the grayscale factor is 1.8, where B_1、G_1 and R_1 are the mean values of the *Camellia oleifera* fruit area, B_2、G_2 and R_2 are the mean values of the background area, and X is the grayscale factor. The binary graph of the *Camellia oleifera* fruit was obtained by the threshold treatment. As can be observed from Figure 6, the HSV color space method is better than the gray factor method in segmenting *Camellia oleifera* fruit, and the segmented binary image of *Camellia oleifera* fruit is more complete. The image segmentation diagram is shown in Figure 6.

$$
\begin{cases}
X \times B_1 - G_1 - R_1 \geq 0 \\
X \times B_2 - G_2 - R_2 \leq 0
\end{cases}
\tag{20}
$$
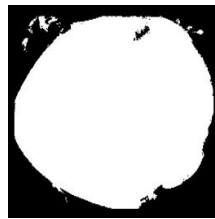
(**a**) Based on HSV color space (**b**) Based on gray scale factor

Figure 6. Image segmentation.

4.3. Morphological Treatment

After image segmentation, the *Camellia oleifera* fruit images can be observed as noise and holes in the tail, which makes subsequent operations difficult. Therefore, morphological processing is carried out for binary images segmented based on color features [33], which mainly includes open and closed operations. Morphological opening operations can eliminate non-target free white spots in the image, and the morphological closing operations can close the internal holes in the binary image of *Camellia oleifera* fruit and achieve accurate contour fitting. Figure 7 shows the morphological processing diagram.

(**a**) Close operation (**b**) Open operation

Figure 7. Morphological processing.

4.4. Centroid of Camellia Oleifera Fruit

Camellia oleifera fruit image processing after morphological processing is median filtered to remove noise; we smooth the edges of the binary image, and make its contour fitting effect better [34]. The mask operation extracts the *Camellia oleifera* fruit part of the target region. Therefore, the mask extraction operation is performed via a binary image obtained after morphological processing and an image with a 10-pixel expansion of image preprocessing. Edge detection is the process of locating the edge pixels of the *Camellia*

oleifera fruit, so the Canny operator is used to detect the edge of the masked image to obtain the outline edge of the *Camellia oleifera* fruit.

After edge detection, we need to use BLOBs to find the centroid of *Camellia oleifera* fruit. BLOB is a group of connected pixels with the same attributes in the image. The moment is used to calculate the center of gravity, area, and other shape features of the shape. The calculation of the image's geometric moment is

$$M_{ji} = \sum_{x,y} I(x,y)x^j y^i \tag{21}$$

Equation (21) represents the pixel value at the pixel. Use moments in OpenCV to find the center of BLOB, that is, the centroid of the *Camellia oleifera* fruit, as shown in Figure 8. The centroid calculation includes Equations (22) and (23):

$$\bar{x} = \frac{m_{10}}{m_{00}} \tag{22}$$

$$\bar{y} = \frac{m_{01}}{m_{00}} \tag{23}$$

where $\bar{x}$ and $\bar{y}$ are the centroid coordinate, m_{00} is the zero-order moment, m_{10} and m_{01} are the first order moments.

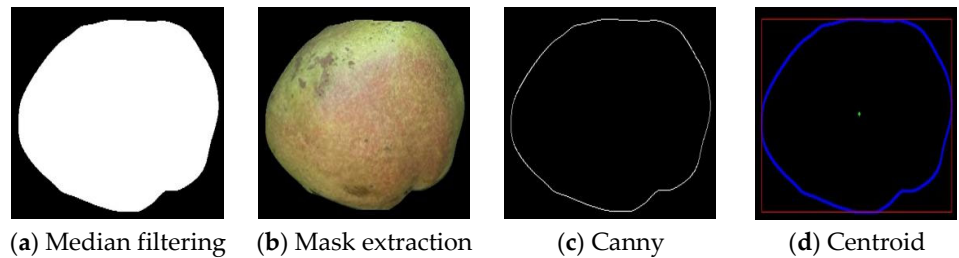(**a**) Median filtering　　(**b**) Mask extraction　　　　(**c**) Canny　　　　　　(**d**) Centroid

Figure 8. Centroid of *Camellia oleifera* fruit.

5. Test Results and Analysis

5.1. Algorithm Evaluation Indicators

The *precision* (P), *recall* (R) and *mean average precision* (*mAP*) are used as evaluation indicators [35]. The specific equations are

$$P = \frac{TP}{TP + FP} \tag{24}$$

$$R = \frac{TP}{TP + FN} \tag{25}$$

$$AP_i = \int_0^1 P(R)d(R) \tag{26}$$

$$mAP = \frac{1}{N}\sum_{i=1}^{N} AP_i \tag{27}$$

TP is the number of positive samples accurately detected, *FP* is the number of negative samples incorrectly detected, and *FN* is the number of positive samples incorrectly detected.

5.2. Algorithm Training Results

First, the input is divided into a training set and validation set into the YOLOv7 and YOLOv5s algorithm networks for training. After 200 epochs of training, the positioning

loss and confidence loss function value curves during training are shown in Figure 9, including the detection box loss and detection object loss. The abscissa represents the number of iterations, and the ordinate represents the loss value.
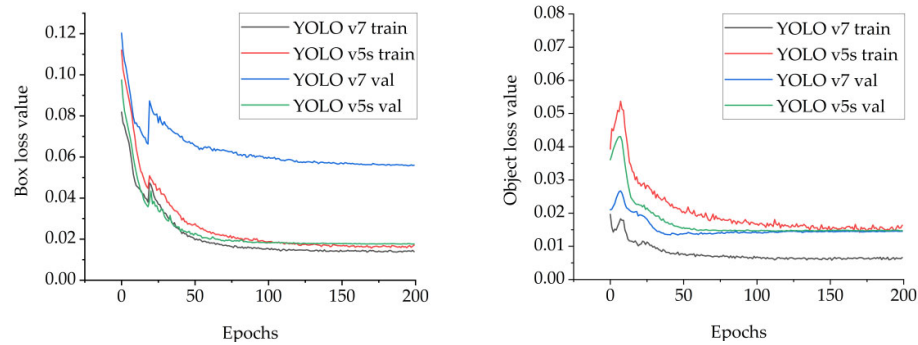

Figure 9. Loss function value curve.

The box loss curve shown in Figure 9 shows that the box loss value of the YOLOv7 algorithm decreases rapidly between 0 and 50 of the training batch, and then the decline rate becomes slow. After 200 epochs of training, the box loss value of YOLOv5s algorithm on the training set is slightly larger than the box loss value of the YOLOv7 algorithm. The box loss value of the YOLOv7 algorithm on the validation set is greater than that of the YOLOv5s algorithm, and finally stabilizes at around 0.06. The object loss curve shown in Figure 9 demonstrates that the object loss value of the YOLOv7 algorithm decreases during training, and the YOLOv5s algorithm shows a trend of rising first and then declining. The object loss value of the YOLOv7 algorithm is less than that of the YOLOv5s algorithm on the validation set, and the final loss value tends to be stable at about 0.015.

5.3. Comparison Test Results and Analysis

To further verify the effectiveness of the YOLOv7 algorithm in the detection of *Camellia oleifera* fruit, under the same experimental conditions, it is compared with the YOLOv5s algorithm target detection algorithm. The experiment takes *precision*, *recall*, and *mean average precision* as the evaluation indicators of the algorithm. The performance comparison results of different detection algorithms are shown in Figure 10 and Table 1.
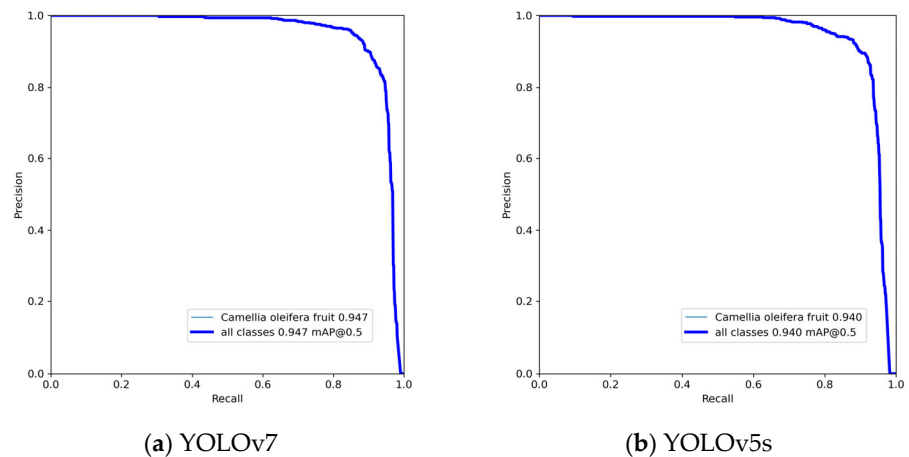

(**a**) YOLOv7   (**b**) YOLOv5s

Figure 10. Precision-recall curves graphs.

Table 1. Performance comparison of different detection algorithms.

Algorithm	$P(\%)$	$R(\%)$	$mAP_{@0.5}(\%)$	$mAP_{[@0.5:0.95]}(\%)$
YOLOv7	92.9	88.3	94.7	83.1
YOLOv5s	93.1	88	94	79

Table 1 shows that the *precision* of YOLOv7 is 92.9%, and that of YOLOv5s is 93.1%. The *precision* of YOLOv7 is basically like that of YOLOv5s. The *mAP@0.5* of YOLOv7 is 94.7%, and *mAP@0.5* of YOLOv5s is 94%. The *mAP@0.5* of YOLOv7 is higher than that of YOLOv5s.

5.4. Comparison of Detection Results

To demonstrate the detection effect of the YOLOv7 algorithm on *Camellia oleifera* fruit, the random images of *Camellia oleifera* fruit taken at the sampling site were detected and compared with the YOLOv5s network. The detection results are shown in Figure 11. On the same data set, YOLOv7 detection takes 15.215s, and YOLOv5s detection takes 16.774s, indicating that YOLOv7 has more advantages in speed.

Original image 1   YOLOv7 (0.97,0.95,0.81)   YOLOv5s (0.96,0.94,0.76)

Original image 2   YOLOv7 (0.98,0.96)   YOLOv5s (0.97,0.94)

| Original image 3 | YOLOv7 (0.97,0.90,0.64) | YOLOv5s (0.96,0.84,0) |

Figure 11. Comparison of detection results.

Figure 11 shows that some *Camellia oleifera* fruits with occlusion can still be detected based on the YOLOv7 algorithm. Compared with the YOLOv5s algorithm, it significantly improves the detection of the usually undetected *Camellia oleifera* fruits.

5.5. Positioning Deviation

After the *Camellia oleifera* fruit is detected by the YOLOv7 algorithm, we need to further determine the location of the picking point. A classical image processing algorithm and mathematical geometry are used to locate the picking point. In this work, 10 sample images are selected when picking a location point. A centroid calculation method can be used, and then the position deviation between the centroid and the center of the YOLO recognition frame is statistically calculated. A total of 10 sample images are used for the experiment that determines the point positioning error at the centroid of *Camellia oleifera* fruit. Table 2 shows that the pixel error value of the image is below 10.

Table 2. Positioning deviation.

NO	The Center of the YOLO Recognition Frame	Centroid	Deviation (Pixel)
1	(199,116)	(205,116)	6
2	(227,114)	(225,110)	4.5
3	(199,81)	(194,80)	5.1
4	(203,92)	(206,92)	3
5	(83,53)	(84,54)	1.4
6	(68,40)	(66,40)	2
7	(222,100)	(221,99)	1.4
8	(132,81)	(132,80)	1
9	(71,34)	(73,34)	2
10	(209,107)	(210,105)	2.2
Average value (pixel)			2.86
Maximum (pixel)			6

The analysis in Table 2 demonstrates that the average deviation position of the image extraction centroid point and YOLO identification frame center point is 2.86; the maximum deviation is 6. According to the statistical data, the position deviation of *Camellia oleifera* fruit detected in this study is relatively small. When the *Camellia oleifera* fruit out-

line accuracy is lower, the positioning accuracy of the *Camellia oleifera* fruit may be relatively large; the pixel difference also affects the actual distance measurement. When the image resolution is lower, the difference between the pixel distance and the actual distance is larger. In general, this method can accurately locate the centroid of *Camellia oleifera* fruit.

6. Conclusions

To detect *Camellia oleifera* fruits in complex environments and solve the problem of the accurate positioning of *Camellia oleifera* fruit picking robots, this research proposes a method based on the fusion of deep learning/visual perception algorithms and classic image processing, which can adaptively and actively locate the *Camellia oleifera* fruit recognition and picking points.

After the testing model and method performance, the following conclusions are obtained:

(1) The *precision* of YOLOv7 and YOLOv5s for model performance is basically similar, and the *mean average precision* of YOLOv7 is higher than that of YOLOv5s.
(2) On the same dataset, YOLOv7 takes less time to detect images than YOLOv5s does. Compared with the YOLOv5s algorithm, the YOLOv7 algorithm can detect some occluded *Camellia oleifera* fruits, which significantly improves *Camellia oleifera* fruit detection.
(3) The pixel value of the position deviation between the image extracted centroid point and the center of the YOLO recognition frame obtained by image processing is less than 10. Thus, the center of the YOLO recognition frame is approximately consistent with the image-extracted centroid point.

Although the YOLOv7 network is applied to *Camellia oleifera* fruit detection and has achieved good results, the detection accuracy still needs to be improved. In future research, we will further optimize the network model structure and continue to improve the identification and detection performance and missing detection performance of the *Camellia oleifera* fruit model. In addition, when designing the *Camellia oleifera* fruit picking robot, it is necessary to consider the problem of the pixel fault tolerance value of the end mechanism to compensate for the pixel error. As the *Camellia oleifera* fruit recognition and detection has to be realized during picking, the influence of camera motion on the detection accuracy needs to be studied subsequently. In addition, the possibility of using multiple cameras to better position the objects also needs to be studied.

Conflicts of Interest: The authors declare that they have no conflict of interest.

References

1. Tang, N.; Tan, X.; Cai, Y.; He, M.Y.; Xiang, Z.Y.; Ye, H.; Ma, J.L. Characterizations and application potentials of the hemicelluloses in waste oil-tea camellia fruit shells from Southern China. *Industrial Crops & Products*, **2022**, 178.
2. Jiang, H.Z.; Hu, Y.L.; Jiang, X.S.; Zhou, H.P. Maturity Stage Discrimination of Camellia oleifera Fruit Using Visible and Near-Infrared Hyperspectral Imaging. *Molecules*, **2022**, 27(19).
3. Yu, L.Y.; Xiong, J.T.; Fang, X.Q.; Yang, Z.G.; Chen, Y.Q.; Lin, X.Y.; Chen, S.F. A litchi fruit recognition method in a natural environment using RGB-D images. *Biosystems Engineering*, **2021**, 204.
4. Kang, H.W.; Zhou, H.Y.; Wang, X.; Chen, C. Real-Time Fruit Recognition and Grasping Estimation for Robotic Apple Harvesting. *Sensors*, **2020**, 20(19).
5. Li, J.H.; Tang, Y.C.; Zou, X.J.; Lin, G.C.; Wang, H.J. Detection of Fruit-Bearing Branches and Localization of Litchi Clusters for Vision-Based Harvesting Robots. *IEEE Access*, **2020**, 8.
6. Chen, M.Y.; Tang, Y.C.; Zou, X.J.; Huang, K.Y.; Huang, Z.F.; Zhou, H.; Wang, C.L.; Lian, G.P. Three-dimensional perception of orchard banana central stock enhanced by adaptive multi-vision technology. *Computers and Electronics in Agriculture*, **2020**, 174(C).
7. Wu, F.Y.; Duan, J.L.; Chen, S.Y.; Ye, Y.X.; Ai, P.Y.; Yang, Z. Multi-target recognition of bananas and automatic positioning for the inflorescence axis cutting point. *Frontiers in Plant Science*, **2021**, 12.
8. Lin, G.C.; Tang, Y.C.; Zou, X.J.; Xiong, J.T.; Fang, Y.M. Color-, depth-, and shape-based 3D fruit detection. *Precision Agriculture: An International Journal on Advances in Precision Agriculture*, **2020**, 21(1).
9. Santos, T.T.; de Souza, L.L.; dos Santos, A.A.; Avila, S. Grape detection, segmentation, and tracking using deep neural networks and three-dimensional association. *Computers and Electronics in Agriculture*, **2020**, 170(C).
10. Xu, D.Q.; Chen, L.T.; Mou, X.W.; Wu, Q.; Sun, G.Q. 3D Reconstruction of Camellia Oleifera Fruit Recognition and Fruit Branch based on Kinect Camera. *2021 2nd International Conference on Artificial Intelligence and Information Systems*. **2021**: 1-6.
11. Gao, F.F.; Fu, L.S.; Zhang, X.; Majeed, Y.; Li, R.; Karkee, M.; Zhang, Q. Multi-class fruit-on-plant detection for apple in SNAP system using Faster R-CNN. *Computers and Electronics in Agriculture*, **2020**, 176.
12. Tang, Y.C.; Chen, M.Y.; Wang, C.L.; Luo, L.F.; Li, J.H.; Lian, G.P.; Zou, X.J. Recognition and Localization Methods for Vision-Based Fruit Picking Robots: A Review. *Frontiers in Plant Science*, **2020**, 11.
13. Xiong, Y.; Peng, C.; Grimstad, L.; From, P.J.; Isler, V. Development and field evaluation of a strawberry harvesting robot with a cable-driven gripper. *Computers and Electronics in Agriculture*, **2019**,*157*: 392-402.
14. Zhang, Y.C.; Zhang, W.B.; Yu, J.Y.; He, L.Y.; Chen, J.N.; He, Y. Complete and accurate holly fruits counting using YOLOX object detection. *Computers and Electronics in Agriculture*, **2022**, 198.
15. Ji, W.; Pan, Y.; Xu, B.; Wang, J.C. A Real-Time Apple Targets Detection Method for Picking Robot Based on ShufflenetV2-YOLOX. *Agriculture*, **2022**, 12(6).
16. Wang, Y., Yan, G.; Meng, Q.L.; Yao, T.; Han, J.F.; Zhang, B. DSE-YOLO: Detail semantics enhancement YOLO for multi-stage strawberry detection. *Computers and Electronics in Agriculture*, **2022**, 198.
17. Tian, Y.N.; Yang, G.D.; Wang,Z.; Wang, H.; Li, E.; Liang, Z.Z. Apple detection during different growth stages in orchards using the improved YOLO-V3 model. *Computers and Electronics in Agriculture*, **2019**, 157.
18. Koirala, A., Walsh, K. B.; Wang, Z.; McCarthy, C. Deep learning for real-time fruit detection and orchard fruit load estimation: benchmarking of 'MangoYOLO'. *Precision Agriculture*, **2019**, 20(6).
19. Gao, F.F.; Fang, W.T.; Sun, X.M.; Wu, Z.C.; Zhao, G.A.; Li, G., Li, R.; Fu, L.S.; Zhang, Q. A novel apple fruit detection and counting methodology based on deep learning and trunk tracking in modern orchard. *Computers and Electronics in Agriculture*, **2022**, 197.
20. Parico A.I.B.; Ahamed, T. Real Time Pear Fruit Detection and Counting Using YOLOv4 Models and Deep SORT. *Sensors*, **2021**, 21(14).
21. Mirhaji, H.; Soleymani M.; Asakereh, A.; Abdanan, M. Fruit detection and load estimation of an orange orchard using the YOLO models through simple approaches in different imaging and illumination conditions. *Computers and Electronics in Agriculture*, **2021**, 191.
22. MacEachern, C.B., Esau, T.J.; Schumann, A.W.; Hennessy, P.J.; Zaman, Q.U. Detection of fruit maturity stage and yield estimation in wild blueberry using deep learning convolutional neural networks. *Smart Agricultural Technology*, **2023**, 3.
23. Tang, Y.C.; Zhou, H.; Wang, H.J.; Zhang, Y.Q. Fruit detection and positioning technology for a Camellia oleifera C. Abel orchard based on improved YOLOv4-tiny model and binocular stereo vision. *Expert Systems With Applications*, **2023**, 211.
24. Lv, J.D.; Xu, H.; Han, Y.; Lu, W.B.; Xu, L.M.; Rong, H.L.; Yang, B.; Zou, L.; Ma, Z.H. A visual identification method for the apple growth forms in the orchard. *Computers and Electronics in Agriculture*, **2022**, 197.
25. Wang, D.D.; He, D.J. Channel pruned YOLO V5s-based deep learning approach for rapid and accurate apple fruitlet detection before fruit thinning. *Biosystems Engineering*, **2021**, 210.
26. Yan, B.; Fan, P.; Lei, X.Y.; Liu, Z.J.; Yang, F.Z. A Real-Time Apple Targets Detection Method for Picking Robot Based on Improved YOLOv5. *Remote Sensing*, **2021**, 13(9).

27. Wu, F.Y.; Duan, J.L.; Ai, P.Y.; Chen, Z.Y.;Yang, Z.; Zou, X.J. Rachis detection and three-dimensional localization of cut off point for vision-based banana robot. *Computers and Electronics in Agriculture*, **2022**, 198.

28. Wang, C.Y.; Bochkovskiy, A.; Liao, H. YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. *arXiv e-prints*, **2022**.

29. Tang, Y.C.; Huang, Z.F.; Chen, Z.; Chen, M.Y.; Zhou, H.; Zhang, H.X.; Sun, J.B. Novel visual crack width measurement based on backbone double-scale features for improved detection automation. *Engineering Structures* **2023**, 274: 115158.

30. Anam, C.; Naufal, A.; Sutanto, H.; Adi, K.; Dougherty, G. Impact of Iterative Bilateral Filtering on the Noise Power Spectrum of Computed Tomography Images. *Algorithms*, **2022**, 15(10).

31. Ali, N.M.; Rashid, N.K.A.M.; Mustafah, Y.M. Performance Comparison between RGB and HSV Color Segmentations for Road Signs Detection. *Applied Mechanics and Materials*, **2013**, 2622(393-393).

32. Mahmoud, S.F.; Hamed, S. Defect Detection in Fruit and Vegetables by Using Machine Vision Systems and Image Processing. *Food Engineering Reviews*, **2022**, 14(3).

33. Fu, L.S.; Tola, E.; Al-Mallahi, A.; Li, R.; Cui, Y.J. A novel image processing algorithm to separate linearly clustered kiwifruits. *Biosystems Engineering*, **2019**, 183(C).

34. Jiang, S.X.; Zhou, R.G.; Hu, W.W.; Li,Y.C. Improved Quantum Image Median Filtering in the Spatial Domain. *International Journal of Theoretical Physics*, **2019**, 58(7).

35. Chen, Z.Y.; Wu, R.H.; Lin, Y.Y.; Li, C.Y.; Chen, S.Y.; Yuan, Z.N.; Chen, S.W.; Zou, X.J. Plant Disease Recognition Model Based on Improved YOLOv5. *Agronomy*,**2022**,12(2).