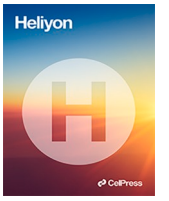
Heliyon 10 (2024) e31868

Contents lists available at ScienceDirect

Heliyon

journal homepage: www.cell.com/heliyon



Research article

A method for cabbage root posture recognition based on YOLOv5s

Fen Qiu [b], Chaofan Shao [a], Cheng Zhou [a], Lili Yao [a,*]

[a] School of Information Engineering, Huzhou University, 313000, Zhejiang, China
[b] Huzhou Academy of Agricultural Sciences, Huzhou, 313000, Zhejiang, China

ARTICLE INFO

2000 MSC:
0000
1111
PACS:
0000
1111
Keywords:
Cabbage root
Object detection
Inclination recognition
Machine vision

ABSTRACT

Efficient, non-destructive cabbage harvesting is crucial for preserving its flavor and quality. Current cabbage harvesting mainly relies on mechanized automatic picking methods. However, a notable deficiency in most existing cabbage harvesting devices is the absence of a root posture recognition system to promptly adjust the root posture, consequently impacting the accuracy of root cutting during harvesting. To address this issue, this study introduces a cabbage root posture recognition method that combines deep learning with traditional image processing algorithms. Preliminary detection of the main root Region of Interest (ROI) areas of the cabbage is carried out through the YOLOv5s deep learning model. Subsequently, traditional image processing methods, the Graham algorithm, and the method of calculating the minimum circumscribed rectangle are employed to specifically detect the inclination angle of cabbage roots. This approach effectively addresses the difficulty in calculating the inclination angle of roots caused by occlusion from outer leaves. The results demonstrate that the precision and recall of this method are 98.7 % and 98.6 %, respectively, with an average absolute error of 0.80° and an average relative error of 1.34 % in posture. Using this method as a reference for mechanical harvesting can effectively mitigate cabbage damage rates.

1. Introduction

With the rapid global development of agricultural mechanization and automation technology, mechanized harvesting has emerged as pivotal approach to augment agricultural production efficiency. Cabbage is a popular green vegetable worldwide, known for its delicious taste, high yield, and resistance to storage and transportation. Harvesting methods for cabbage mainly including manual and mechanical harvesting. While manual harvesting minimizes cabbage damage and ensures high precision in root cutting, it remains time-consuming and labor-intensive [1]. Conversely, mechanical harvesting methods are efficient, cost-effective, and more suitable for large-scale cabbage farming. However, inherent factors such as the natural tilt of cabbage roots or collisions with harvesting machinery during operation may alter the relative positioning of the root and the axis of the cabbage, posing challenges in maintaining alignment with the disc knife of the root cutting mechanism. Consequently, this could lead to inaccurate root cutting and potential cabbage damage [2]. To solve this problem, adjusting the posture of the cabbage before root cutting is important, and precision identification of the cabbage tilt degree is a critical foundation for such adjustment [3].

This study addresses the significant influence of root inclination on the efficiency and quality of mechanized cabbage harvesting, aiming to develop a model that integrates deep learning with image processing techniques for precise and prompt identification of

* Corresponding author.
E-mail address: 03120@zjhu.edu.cn (L. Yao).

https://doi.org/10.1016/j.heliyon.2024.e31868
Received 11 September 2023; Received in revised form 20 May 2024; Accepted 21 May 2024
Available online 29 June 2024

cabbage root inclination. While previous research has acknowledged the effectiveness of both non-image sensors and image sensors in detecting plant tilt, challenges persist in achieving robust stability in dynamic testing scenarios and addressing issues such as leaf occlusion and intricate testing environments. Thus, the fusion of image sensing technology and deep learning models presents heightened potential in terms of recognition accuracy, adaptability, and information extraction capability.

While there are precedents for crop identification using deep learning, reports on methods for cabbage root inclination recognition are scarce. In light of this, this article proposes a cabbage root tilt recognition method that integrates the YOLOv5s model with image processing technology, which significantly improves both the speed and accuracy of cabbage root recognition, leading to enhanced efficiency and quality in mechanized harvesting. The main contributions are summarized as follows:

(1) Based on the combination of the YOLOv5s model and traditional image processing methods, the proposed method achieves rapid and efficient recognition of cabbage root under limited hardware resources, thus meeting the real-time operational requirements of harvesting machinery.
(2) By introducing the Graham algorithm and the method of minimum circumscribed rectangle calculation, the interference of outer leaves on cabbage posture recognition is effectively addressed. This enables precise calculation of the inclination angle of cabbage roots, providing a reference basis for actual harvesting.
(3) The proposed method helps to fill the current gap in research on cabbage root posture recognition, thus providing references and insights for subsequent studies.

The remaining sections of this paper are organized as follows:

Section 2 provides a comprehensive literature review of previous studies that have utilized conventional image processing methods and deep learning techniques for fruit and vegetable pose recognition. In Section 3, the materials and methods are described, including the structure and implementation of the cabbage detection algorithm. Section 4 presents the experimental results and comparisons. Section 5 discusses and analyzes the experimental results and comparisons. Finally, Section 6 concludes the research and outlines future work plans.

2. Literature review

Previous research has shown that both non-image sensors and image sensors can effectively identify plant tilt degrees. Non-image sensors are advantageous for their high accuracy, ease of use, and quick response in practical applications [4,5]. For instance, Bunce et al. developed a biaxial tilt meter using two inclinometers and a data collector to detect slight angle tilts of tree trunks [6]. However, this device requires an external power source, and its test stability will be affected by environmental factors. Park et al. used accelerometers, gyroscopes, and linear potentiometers to collect the cutting height, angle, and position of the cabbage [7]. Based on Kalman filters, they predicted the tilt posture of the cutting device, improving the accuracy of cabbage harvesting. However, the accuracy of these sensors tends to degrade over time. In another approach, James et al. used a triaxial accelerometer to predict the tilt angle of the tree root area by measuring the dynamic and static tension [8]. Guo et al. used strain sensors and two uniaxial angle sensors to measure the tilt angle of corn stalks, assessing the tilt state of corn under applied pressure by utilizing the equivalent force in the vertical direction of the stalk [9]. Although the methods mentioned above realized the measurement of plant tilt preliminary, non-image sensors often require physical contact with the plant during measurements, demanding high stability for both the sensor and the test subject, which is hard to meet the needs of dynamic testing.

Compared to non-image sensors, image sensors have demonstrated remarkable accuracy, adaptability, and information extraction capabilities owing to the rapid advancements in image processing technology, particularly in identifying regular fruit shapes. Features such as contours, colors, and textures are commonly employed for this purpose. Roomi et al. used a Bayes classifier to identify fruit contours by extracting shape and region features, which showed high accuracy when applied to the classification of Indian mangoes [10]. Hannan et al. effectively distinguished oranges from other fruits or objects using shape analysis techniques and machine learning algorithms [11]. Similarly, Wei et al. effectively identified mature fruits by extracting new features from the OHTA color space using an improved Otsu threshold algorithm, which performs well under complex backgrounds [12]. Rakun et al. used a texture segmentation method based on spatial frequency distribution to identify different fruits, overcoming the problems of uneven illumination, fruit occlusion, and similar background features in complex environments by clustering pixels with similar texture features [13]. Furthermore, Wu et al. proposed an automatic tomato maturity identification algorithm that combines multi-feature fusion, feature analysis and selection, a weighted Relevance Vector Machine (RVM) classifier, and a two layer classification strategy, which achieved a detection accuracy of 94.90 % on 120 images [14]. These studies illustrate that relying on either single features or feature fusion can effectively identify regular fruit shapes. However, these methods encounter limitations in identifying irregular fruit shapes and addressing occlusion issues during identification. Given the intricate morphology and posture of cabbage roots, often obscured by outer leaves, traditional image processing methods alone cannot accurately identify and locate the root, let alone measure its tilt degree. Therefore, the introduction of more precise algorithms and techniques is imperative to address this challenge.

Existing research has shown that deep learning algorithm models, such as Convolutional Neural Networks (CNN) and Deep Convolutional Neural Networks (DCNN), have higher recognition accuracy for irregular fruits compared to traditional image processing methods [15]. This heightened accuracy can be attributed to their capability to autonomously learn target features, thereby showcasing robustness against target occlusion and variations in lighting conditions [16]. Momeny et al. successfully identified and classified the appearance of cherries with regular and irregular shapes by improving the CNN algorithm [17]. Yan et al. proposed an improved Faster R–CNN algorithm that successfully identified irregularly shaped prickly pear fruit, achieving an average recognition

accuracy of 94.6 % [18]. Wang et al. improved the Faster R–CNN model for detecting multiple targets of tomato maturity under complex scenarios of branch occlusion, fruit overlap, and light influence [19]. The improved model outperformed other common object detection models, with the average accuracy (mAP) of tomato maturity recognition reaching up to 96.14 %.

Furthermore, the YOLO (You Only Look Once) model has garnered significant attention across various domains due to its exceptional real-time recognition capabilities. Its applicability extends to diverse fields, such as plantar pressure angle detection and recognition [20], diagnosis of Alzheimer's disease [21], detection of lumbar disc protrusion [22], among others. These applications highlight the immense potential of the YOLO model in addressing complex recognition challenges. In the field of agriculture, Tian et al. improved the YOLOv3 model to detect apples in different growth stages under changing light conditions in orchards [23]. Their method facilitated real-time and effective detection of overlapping apples and those under obfuscating conditions. Ji et al. accurately identified and located apples in various complex environments using an improved lightweight YOLOv4 method, which also effectively enhanced the level of lightness of the model while improving recognition speed [24]. Wang et al. improved the algorithm based on YOLOv5 for real-time identification of apple stem. The optimized YOLOv5 achieved an accuracy of 93.89 % in detecting apple stem of different varieties [25]. Xie et al. utilized the YOLOv7 target detection model to achieve efficient feature extraction and accurate target detection of pear calyx and stem. The test results demonstrate that the proposed model achieves high detection accuracy while maintaining low computational complexity and parameter count [26]. Dewi et al. employed the YOLOv8 target detection model to distinguish and classify various fruit varieties, achieving a detection accuracy as high as 0.97 % [27]. These models exhibit robustness against target occlusion and variations in lighting conditions, delivering exceptional performance in terms of recognition accuracy and real-time capability. However, due to problems such as leaf occlusion, complex testing environments, and poor consistency, existing algorithms cannot be effectively transplanted into research on cabbage root tilt detection. In addition, mechanical cabbage harvesting requires high recognition speeds. While ensuring recognition accuracy, it is necessary to improve the recognition speed as much as possible to meet the basic needs of real-time root cutting.

3. Materials and methods

3.1. Overview of the cabbage root posture recognition process

The proposed cabbage root posture recognition algorithm workflow is shown in Fig. 1 and can be delineated into the following steps:

Step 1. The cabbage image dataset is meticulously annotated, and the organized cabbage images are inputted into the pre-trained

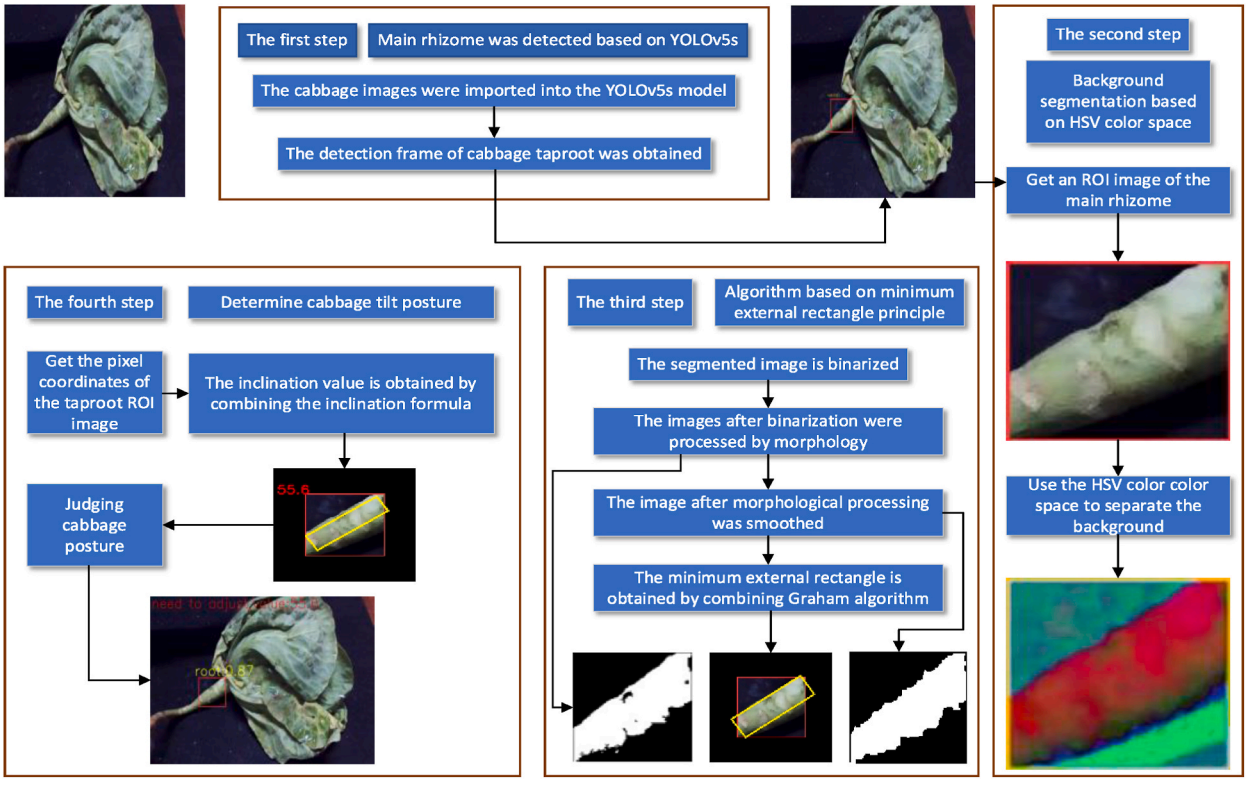


Fig. 1. Workflow of the pose recognition algorithm for cabbage root.

YOLOv5s model. This model employs image feature extraction and target detection to identify the main root of the cabbage. During the detection phase, the model divides the input cabbage image into multiple grid units, then classifies and locates the bounding box for each unit, thereby obtaining the detection box of the cabbage main root.

Step 2. Based on the location information of the cabbage main root detection box, the ROI of the main root is extracted from the original cabbage image and the HSV color threshold method is used to distinguish the background from the main root.

Step 3. The ROI undergoes image segmentation, followed by binarization of the resulting image. The binarized main root image is processed using morphological opening operations to eliminate voids in the main root, remove isolated small blocks, smooth the surface of the main root, and obtain the minimum circumscribed rectangle of the main root in combination with the Grayham algorithm, thus obtaining the four-pixel coordinates of the rectangle.

Step 4. Using the four-pixel coordinates of the main root minimum circumscribed rectangle obtained, calculate the length of the rectangle based on the pixel coordinates and calculate the tilt value using the tilt formula to judge the cutting posture of the cabbage.

3.2. Experimental materials

The cabbages used for the experiment were collected from Dabanqiao Town, Yunnan Province. The chosen variety was Lotus White, and the harvesting took place in August 2022. To ensure freshness, all collected samples were adequately refrigerated for preservation. The experimental equipment includes an electronic angle measuring instrument, an industrial-grade CCD camera, and a computer. Under the illumination of an LED light source, the cabbage was photographed against a black background for image collection. The CCD camera was precisely aimed at the cabbage, and adjustments were made to the object distance and focus to achieve optimal image quality. At the same time, setting fixed parameters is conducive to subsequent image collection and video recording, facilitating rapid and accurate transmission of collected image data to the computer for analysis and processing.

3.3. Experimental environment

In order to ensure data diversity, the cabbages were photographed were photographed from various angles, including the front, back, and side, under different light intensities, this resulted in a total of 1500 original images (Fig. 2 (a)). Subsequently, data augmentation techniques such as horizontal flipping (Fig. 2 (b)), rotation (Fig. 2 (c)), Gaussian blur (Fig. 2 (d)), brightness enhancement (Fig. 2 (e)), and scaling (Fig. 2 (f)) were employed to increase the sample data to 2184 images. The annotation process was conducted using the labelImg software and professional cabbage harvesting personnel are invited to verify the accuracy of the annotation work. The position of the main root of the cabbage in each image was manually identified and annotated. The annotation results were saved as txt annotation files in the YOLO file format, and the annotation process is shown in Fig. 3. The dataset was divided into training, test, and validation sets in a 7:2:1 ratio to provide support for the training and evaluation of the model.

The training process was carried out on a cloud server, its hardware configuration processor is the Inte Xeon Platinum 8255C @2.50 GHz, the GPU is the RTX 2080 Ti (11 GB), the python version is 3.8.12, and the cuda version is 11.3.1. Traditional image processing was conducted using the OpenCV-python, Numpy, and Matplotlib libraries, and the construction of the YOLOv5 model

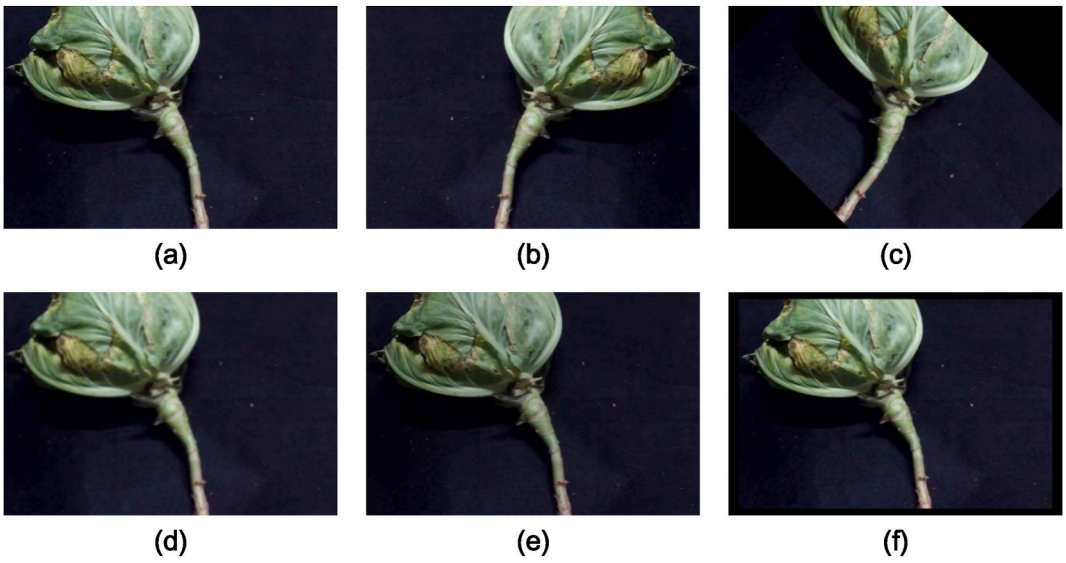


Fig. 2. Image enhancement rendering. (a). Original image, (b). Flip horizontal, (c). Rotated image, (d). Image with Gaussian blur applied, (e). Image after brightness enhancement, (f). Resized image.

Fig. 3. LabelImg software annotation process.

referred to the officially released code.

3.4. Cabbage root detection algorithm

3.4.1. YOLOv5s network architecture

Currently, various deep learning algorithm models can be employed for object detection, including the most common ones are the R–CNN [28–30] series, YOLO [31,32] series, and SSD [33] series, etc. Each of these models possesses distinct advantages and limitations, and the choice should be guided by practical application requirements. The R–CNN series models, including Fast R–CNN, Faster R–CNN, etc., are renowned for their high accuracy and have found widespread use. However, these models need to perform steps such as region and feature extraction during the training process, so they have higher demands on computational performance and may not be able to meet the requirements for real-time operation. In contrast, the YOLOv5 model has high detection accuracy and relatively low demands on hardware computing capabilities. In addition, YOLOv5 also supports running on different sizes of network architectures, making it possible to achieve good performance on limited hardware resources.

While the YOLO series has progressed to YOLOv8, YOLOv5 remains a prevalent choice due to its stability and maturity. Since its launch, YOLOv5 has been recognized for its reliability and performance efficiency, and its codebase, documentation, and community support are mature enough for developers to learn and use. In addition, YOLOv5 has shown good compatibility and deployment performance on a variety of platforms, which is particularly important for the project of deploying models in cabbage harvester in complex environments. In summary, YOLOv5 was chosen as the object detection model for this study owing to its myriad advantages, including stability, ease of use, strong community support, balanced performance efficiency, and versatile compatibility and deployment capabilities.

The network architecture of YOLOv5s consists of four primary components: Input, Backbone, Neck, and Head.

1. Input: By combining four different training images into a large image, more object instances and background changes are displayed in a single image, so as to enhance the ability of the model to recognize small objects in complex scenes and provide rich context information and object size changes, adaptive anchor box calculation is used to find the best anchor box values in different training sets, and adaptive image scaling is used to reduce the amount of computation, thereby improving target detection speed.
2. Backbone: The Backbone consists of the Focus structure and Cross Stage Partial (CSP) network structure. The Focus structure uses slicing operations to split the high-resolution feature map into multiple low-resolution feature maps, which can reduce the information loss caused by downsampling. The CSP structure splits the original input into two branches, one for convolution operations to reduce the number of channels, and the other for multiple Bottleneck operations, and finally keeps the input and output size consistent through concat, so that the model can learn more features.
3. Neck: The Neck component utilizes the Feature Pyramid Network (FPN) and Path Aggregation Network (PAN) structure. FPN [34] upsamples from top to bottom, allowing lower-level features to gain stronger semantic information; PAN downsamples from bottom to top, making the top-level features contain stronger feature information. By integrating these two features, it can ensure that feature maps of different sizes all contain semantic and feature information, thereby enhancing accurate predictions of images.

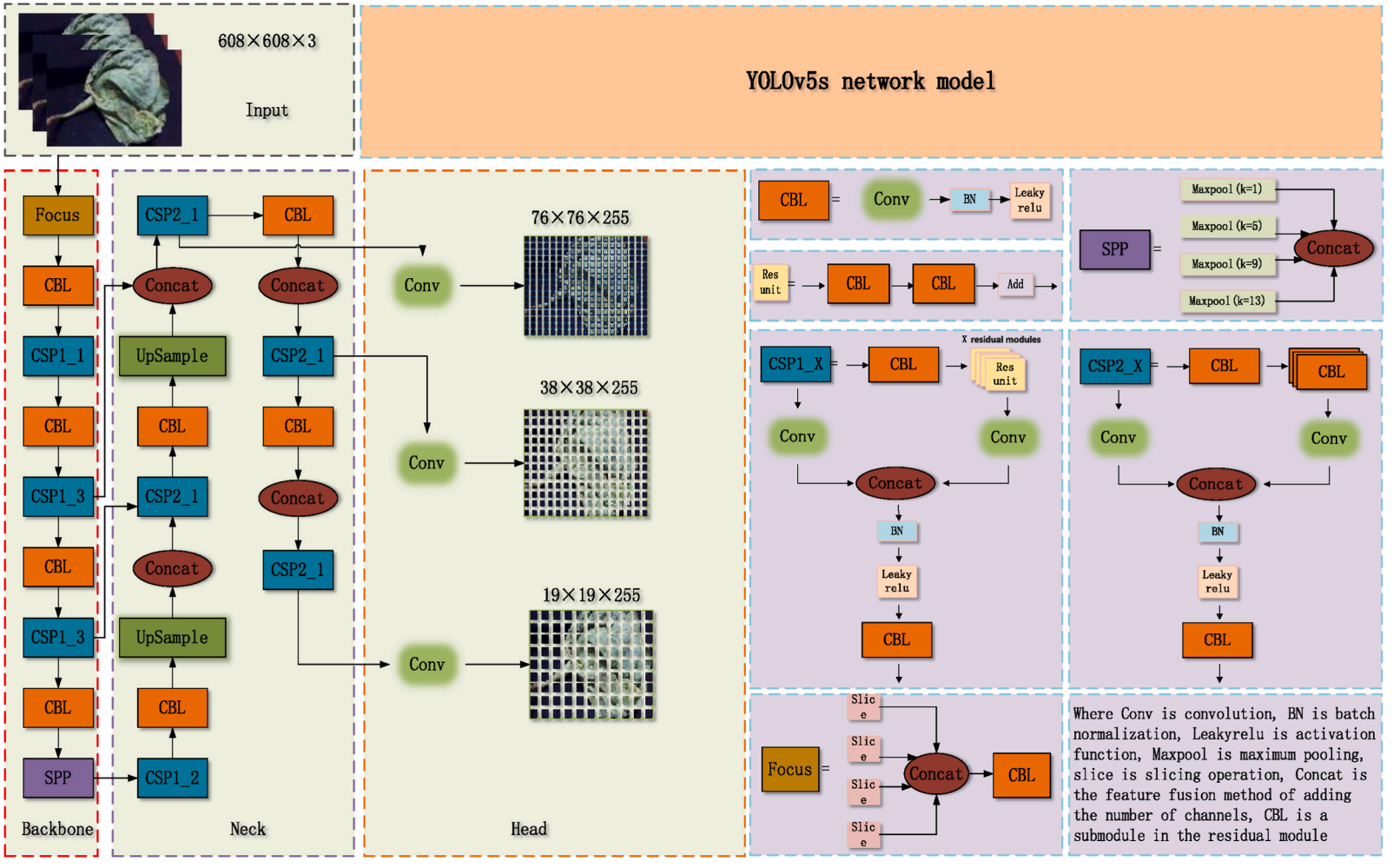


Fig. 4. Yolov5s network model framework.

4. Head: This component includes three Detect detectors, using anchors to perform object detection on feature maps of different scales, obtaining the category and location information of the target.

As shown in Fig. 4, with the above architecture and characteristics, the YOLOv5s network can effectively recognize the cabbage root.

3.4.2. Cabbage recognition

During cabbage recognition, a color background threshold segmentation method based on HSV space is used to perform background segmentation of the main root ROI image of the cabbage. Given the significant color difference between the cabbage root and other parts of the harvester, the YOLOv5s model is used to detect the ROI area of the cabbage root, as shown in Fig. 5 (a). Once the main root ROI is determined (Fig. 5 (b)), this image is transformed from RGB space to HSV space (Fig. 5 (c)). Through multiple experiments, this study has determined the thresholds for hue (H), saturation (S), and brightness (V) that can best achieve segmentation. After completing the above operations, binary calculation is performed to determine the main root position, as shown in Fig. 5 (d).

After determining the position of the main root of the cabbage, cabbage posture recognition is carried out based on the principle of the minimum circumscribed rectangle. The specific algorithm flow is outlined as follows:

1. Using YOLOv5s for object detection, define the set of points on the detection box at the vertex of the object contour as $S = \{r_1, r_2, \ldots, r_n\}$, where $r_i = (x_i, y_i)$. If multiple points share the same y-coordinate, choose the point with the smallest x-coordinate as the starting point. Then traverse all points, and determine whether the y-coordinate of the current point is smaller than that of the starting point, or whether the y-coordinate of the current point is equal to the that of the starting point but the x-coordinate is smaller than the x-coordinate of the starting point. If these conditions are met, the current point is designated as the new starting point, r_{min}.
2. According to the Graham algorithm, designate the r_{min} obtained in step (1) as the origin P_0, and connect to the point P_1 to form the line segment L_0, connect to other points respectively and put the obtained line segments into the line segment set L, then calculate the angle between the connecting line segment and the x-coordinate, and arrange the points in the set P from small to large according to the size of the angle and the length of the line segment, and put them into the point set A.
3. Connect A_{i-1} and A_{i+1} to get the line segment L_i, determine whether the current point A_i lies to the left or right of the line segment L_i. If it lies to the right, then A_{i+1} is not the desired point, and thus it is removed from the point set A; Conversely, if it lies to the left (or on the line), it is the point we are searching for, and thus it is retained in the point set A. These operations are iterated until the condition is met, ultimately resulting in the configuration depicted in Fig. 6 (a).
4. Traverse the point set P(x,y), where (x,y) represents the pixel position in the binary image, as shown in Eq. (1):

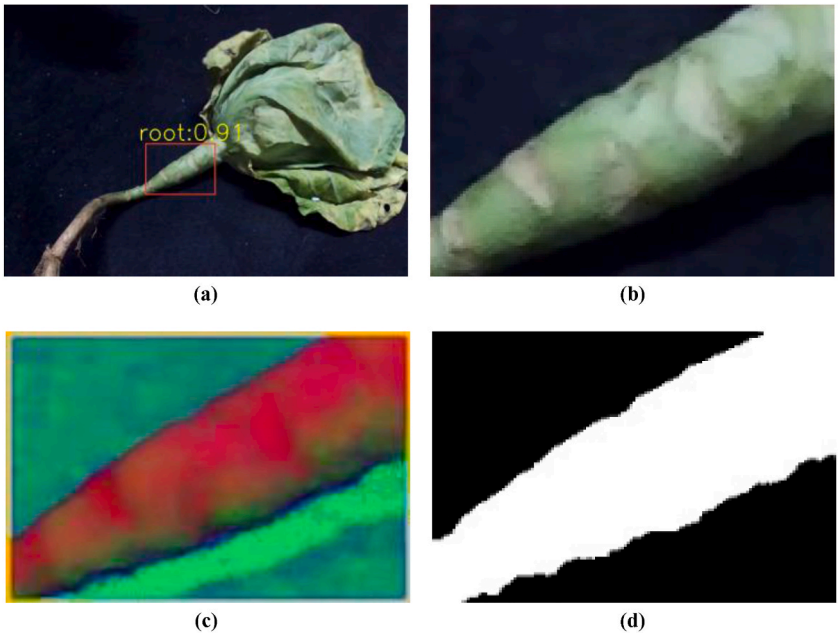


Fig. 5. The Color background thresholding step is based on HSV color space. (a). YOLOv5 detection, (b). ROI location, (c). HSV color space, (d). Binary picture. (For interpretation of the references to color in this figure legend, the reader is referred to the Web version of this article.)

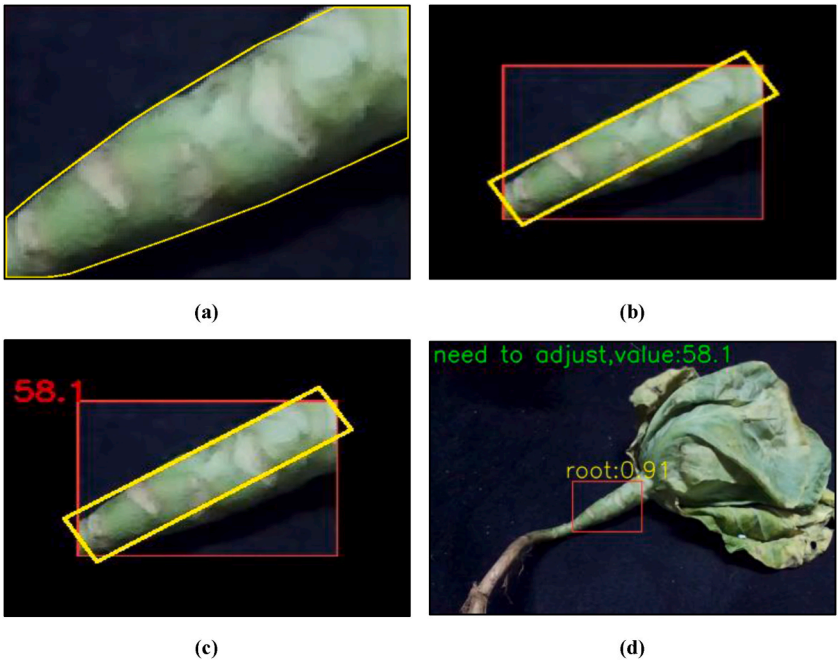


Fig. 6. The identification step is based on the principle of minimum circumscribed rectangle. (a). Convex hull, (b) Minimum enclosing rectangle, (c). Value of skew, (d). Judgment value.

$$\begin{aligned} M_{00} &= \sum_x \sum_y P(x,y) \\ M_{10} &= \sum_x \sum_y x \times P(x,y) \\ M_{01} &= \sum_x \sum_y y \times P(x,y) \end{aligned} \tag{1}$$

where M_{00} represents the sum of the binary pixel values within the image, indicating the total number of pixels with a value. Additionally, M_{10} and M_{01} denote the weighted sums of the horizontal and vertical coordinates of the image, respectively, where the weights correspond to the pixel coordinates within the image. Consequently, we derive the following Eq. (2):

$$\begin{cases} X_c = \dfrac{M_{10}}{M_{00}} \\ Y_c = \dfrac{M_{01}}{M_{00}} \end{cases} \tag{2}$$

From this, the centroid coordinates are obatained $O(X_c,Y_c)$.

5. Find the maximum and minimum points X_{min}, X_{max}, Y_{min}, Y_{max}, within the horizontal and vertical coordinates of the point set A. Subsequently, construct the initial minimum circumscribed rectangle R_0 with the centroid coordinates O, and record the area AreaMin and the positions of the four points.
6. Rotate the calculated AreaMin rectangle R_1 by an angle α to get the final minimum circumscribed matrix replacing the original R_0, and calculate and obtain the coordinates of the four vertices of the new rectangle R_1. The final minimum circumscribed rectangle is obtained, as shown in Fig. 6 (b).
7. Arrange the coordinates of the obtained four vertices into the set $V = (m_i, n_i)$ $(0 \le i \le 4, i \in Z)$, according to the distance formula, and calculate the distance D (Eq. (3)) and angle θ (Eq. (4)) between two points. Find the maximum, and then use coordinates of the corresponding two points to calculate the angle θ of the cabbage root. As shown in Fig. 6 (c).

$$D = \sqrt{(m_{i+1} - m_i)^2 - (n_{i+1} - n_i)^2} \tag{3}$$

$$\theta = \left| \tan^{-1}\left(\frac{n_{1+1} - n_1}{m_{1+1} - m_1}\right) \right| \tag{4}$$

8. Compare the calculated angle θ with the cutting threshold ε to assess the posture of the cabbage, the calculation formula is shown as Eq. (5). If θ is less than or equal to ε, then the cabbage is deemed to be in a normal state, and cutting operations can be performed. If θ exceeds ε, indicating cabbage tilt, adjustments are required before cutting operations can commence. This process is shown in Fig. 6 (d).

$$\text{Cabbage Posture} = \begin{cases} \text{normal}, & \text{if } \theta \le \epsilon \\ \text{inclined}, & \text{if } \theta > \epsilon \end{cases} \tag{5}$$

3.5. Evaluation metrics

3.5.1. Evaluation metrics for object detection models

The evaluation metrics for object detection models include Precision, Recall, F1-score, and mean Average Precision (mAP). Precision reflects the proportion of actual positive samples in the model prediction for positive samples, while Recall shows the proportion of actual positive samples correctly predicted by the model. The F1-score is the harmonic mean of Precision and Recall, which can comprehensively evaluate the performance of the model, avoiding focusing only on single Precision or Recall. In object detection tasks, models with high Recall and low Precision may lead to missed detection of targets, while models with high Precision and low Recall may detect non-target objects by mistake. The calculation formulas for Precision, Recall, and F1 score are shown in Eqs. (6)–(8):

$$\text{Recall} = \frac{TP}{TP + FN} \tag{6}$$

$$\text{Precision} = \frac{TP}{TP + FP} \tag{7}$$

$$F1 = \frac{2 \times \text{Recall} \times \text{Precision}}{\text{Recall} + \text{Precision}} \tag{8}$$

where TP (True Positive) represents the number of samples that the model correctly predicted as positive samples, False Positive (FP) represents the count of samples inaccurately predicted as positive from negative samples, and False Negative (FN) indicates the count of samples inaccurately predicted as negative from positive samples. In addition, mAP serves as a crucial metric for assessing object detection models, calculated based on the average precision across all recall levels. Specifically, mAP50 is the mAP at an Intersection over Union (IoU) threshold of 0.5. Through these metrics, the performance of the object detection model can be more comprehensively and accurately evaluated.

3.5.2. Evaluation metrics for measurement algorithms

This study uses Mean Absolute Error (MAE), Mean Square Error (MSE), Root Mean Square Error (RMSE), and Mean Absolute Percentage Error (MAPE) metrics to evaluate the accuracy of the measurement algorithm. Among them, MAE represents the average absolute error between the measured tilt value and the actual tilt value. MSE represents the average square error between the measured tilt value and the actual tilt value, emphasizing larger errors. RMSE is the square root of MSE. Due to its dimension and scale being the same as the original data, it is more superior in interpretability. MAPE represents the average percentage error between the measured tilt value and the actual tilt value. It provides a scale-independent error index for comparing various tilt measurement systems, and the smaller metric values represent the higher test accuracy. The calculation formulas for MAE, MSE, RMSE, and MAPE are shown in Eqs. (9)–(12):

$$MAE = \frac{1}{n}\sum_{i=1}^{n}\left|y_r - y_p\right| \tag{9}$$

$$MSE = \frac{1}{n}\sum_{i=1}^{n}\left(y_r - y_p\right)^2 \tag{10}$$

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(y_r - y_p\right)^2} \tag{11}$$

$$MAPE = \frac{1}{n}\sum_{i=1}^{n}\frac{\left|y_r - y_p\right|}{y_r} \tag{12}$$

where, y_r represents the actual tilt value, y_p represents the tilt value measured by the computer, and n represents the number of samples.

4. Results

In order to verify the detection performance of the proposed algorithm, alongside implementing experiments using the method outlined in this article, five classic object detection models were also tested under the same experimental conditions for comparison.

4.1. Model training

During the training process of the YOLOv5s model, the hyperparameters are set as follows: the training period is 100 epochs, with a batch size of 32 and 8 working processes. Images were resized to a resolution of (640,640), and both the initial learning rate and learning steps are set to 0.01. The weight decay coefficient is 0.0005, and the momentum is set to 0.8. Additionally, to maintain model accuracy while reducing the size of the model, comparisons were made among the results data of each weight file provided by the official source, as shown in Table 1, YOLOv5s.pt was finally selected as the pre-trained weights.

The hyperparameters and output indicators during the model training process were automatically recorded through the Wandb tool. This tool can dynamically monitoring the status of model training, perform visualization and result comparison, so as to observe the impact of the number of training iterations on model performance and equipment status, as shown in Fig. 7. It is evident that during the initial 0–20 iterations of the model, box_loss and obj_loss exhibit rapid decreases and convergence, while precision, recall, and mAP50 rapid increases. However, as the iterations progress beyond approximately 20–100, the rates of decrease for box_loss and obj_loss slow down. Meanwhile, the curves representing precision, recall, and mAP50 tend to plateau, ceasing to show significant increases.

4.2. Detection results of cabbage root under different models

YOLOv5s, SSD, Fast R–CNN (FRCN), Adaptive Training Sample Selection (ATSS) and RetinaNet were tested and trained based on the same method and experimental environment mentioned in section 3.3. The test results are shown in Table 2. According to the evaluation indicators of the object detection model in the table, the detection results were compared and analyzed to demonstrate the most optimal detection performance.

According to the findings presented in Table 2, the YOLOv5s model showcased remarkable performance, excelling in key metrics such as Precision, Recall, mAP50, and F1, achieving scores of 99.7 %, 99.7 %, 99.5 %, and 99.7 % respectively. With a mere model size of 14.4 MB, it demonstrated an impressive processing speed of 285 frames per second. On the other hand, the SSD model exhibited commendable Precision and F1 scores of 71.8 % and 83.8 % respectively, but displayed lower performance in Recall and mAP50, scoring 98.1 % and 90.2 % respectively, while processing at a slower rate of 22.4 frames per second. Despite its substantial size, the FRCN model yielded balanced results across Precision, Recall, mAP50, and F1 metrics, achieving scores of 95.1 %, 97.5 %, 90.4 %, and 96.3 % respectively, albeit processing only 10 frames per second. The ATSS model exhibited commendable prowess in Recall and mAP50, acquiring 99.4 % and 90.1 % respectively, but languished in Precision and F1, recording only 12.3 % and 22 %, processing at a rate of 26 frames per second despite its moderate size of 161 MB. The RetinaNet model posted performance metrics of 79.6 %, 98.1 %, 90 %, and 86.7 % in Precision, Recall, mAP50, and F1 respectively, with a substantial size of 256.4 MB, processing 12 frames per second. Upon comprehensive analysis of the six parameters detailed in the table, it is evident that the YOLOv5s model outperformed others in detecting the primary root of the cabbage.

4.3. Comparison of cabbage root tilt tests

In the cabbage main root tilt detection experiment, this article compared the SE-YOLOv5, HT-YOLOv5, and MER-YOLOv5 algorithms based on YOLOv5. SE-YOLOv5 integrates the Skeleton Extraction principle, HT-YOLOv5 incorporates the Hough Transform principle, and MER-YOLOv5 employs the Minimum Circumscribed Rectangle principle, as shown in Table 3.

There exists a discernible disparity in the prediction accuracy between SE-YOLOv5 and HT-YOLOv5. The MAE, MSE, RMSE, and MAPE of SE-YOLOv5 are 6.33°, 68.70°, 8.29, and 31.28 %, respectively, while the corresponding values of HT-YOLOv5 are 5.95°, 62.20°, 7.89, and 39.26 %, respectively. These values indicate that the prediction accuracy of HT-YOLOv5 is higher than SE-YOLOv5 in terms of MAE, MSE, and RMSE, but its error is more significant when dealing with extreme value prediction. In contrast, the performance of MER-YOLOv5 notably outshines the other methodologies. With substantially smaller prediction errors (MAE, MSE, and RMSE) of 1.69°, 4.90°, and 2.21, respectively, and a correspondingly lower MAPE of 15.13 %, MER-YOLOv5 not only achieves high overall prediction accuracy, but also demonstrates smaller errors in predicting extreme values.

Table 1
Comparison of five models with different weights.

Models	Size(pixels)	mAP95 (%)	mAP50 (%)	Params (MB)	FLOPs (B)
YOLOv5n	640	28	45.7	1.9	4.5
YOLOv5s	640	37.4	56.8	7.2	16.5
YOLOv5m	640	45.4	64.1	21.2	49.0
YOLOv5l	640	49.0	67.3	46.5	109.1
YOLOv5x	640	50.7	68.9	86.7	205.7

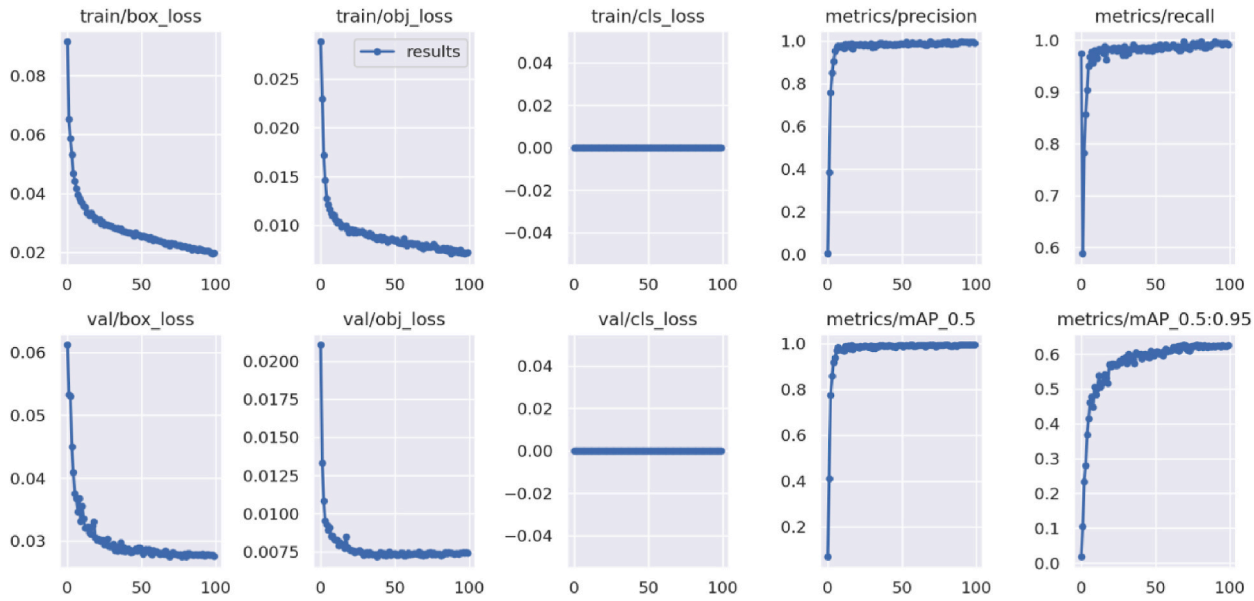


Fig. 7. Analysis of yolov5s results.

Table 2
Comparison test results of different object detection models.

Models	Precision (%)	Recall (%)	mAP50 (%)	F1 (%)	Size (MB)	Fps
YOLOv5s	99.7	99.7	99.5	99.7	14.4	285
SSD	71.8	98.1	90.2	83.8	95	22
FRCN	95.1	97.5	90.4	96.3	377.4	10
ATSS	12.3	99.4	90.1	22	161	26
RetinaNet	79.6	98.1	90	86.7	256.4	12

Table 3
Comparison of the results of different methods for identifying cabbage roots.

Method	MAE (°)	MSE (°)	RMSE	MAPE (%)
SE-YOLOv5	6.33	68.70	8.29	31.28
HT-YOLOv5	5.95	62.20	7.89	39.26
MER-YOLOv5	1.69	4.90	2.21	15.13

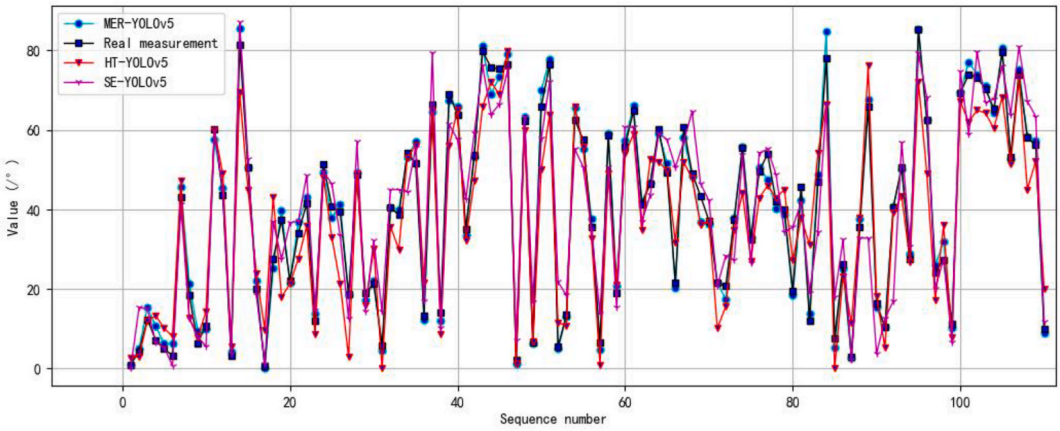


Fig. 8. Numerical plots of angles measured by different algorithms versus manual measurements.

More convincingly, the angle changes measured by the MER-YOLOv5 method is very close to the manually measured tilt angle. In contrast, the angles measured by the SE-YOLOv5 and HT-YOLOv5 methods display greater fluctuations compared to the manually measured angles, as shown in Fig. 8. This serves as additional evidence supporting the accuracy and reliability of the MER-YOLOv5 method.

In conclusion, compared to the SE-YOLOv5 and HT-YOLOv5 methods, the prediction accuracy of the MER-YOLOv5 method is the highest, while the error is the smallest. It is the most suitable prediction method for predicting the tilt of the main root of the cabbage.

5. Discussion

5.1. Implications

Much existing research has conducted adaptive threshold, Canny edge detection, and skeleton extraction methods to deal with the plant root posture recognition problems. For example, Huipeng et al. [35] obtained the approximate outline of the grape stalk through adaptive threshold and Canny edge detection, and finally determined the picking point through the cumulative probability Hough transform. However, these methods could not accurately fit the line based on the root outline after morphological processing, as shown in Fig. 9 (a) and (b). At the same time, the skeleton extraction method proposed by Qi et al. [36], although it can quickly obtain the fitted line, cannot accurately obtain the tilt angle due to the thick root of the cabbage, as shown in Fig. 9 (c) and (d).

In addressing the limitations of the aforementioned methods, this paper presents a cabbage root inclination detection approach based on minimum circumscribed rectangle, which accurately discerns the tilt angle of cabbage, as demonstrated in the earlier measurement outcomes and compared against actual measurements. This method surpasses the constraints of previous approaches when handling cabbages with intricate morphology and thick roots, thereby enabling more precise detection of cabbage posture tilt angles and enhancing prediction accuracy and practicality, as shown in Fig. 9 (e) and (f).

The significance of this study lies in the development of a novel method for detecting the tilt angle of cabbage roots. This approach not only enhances the precision of crop identification and diminishes crop damage during mechanized harvesting but also improves harvest efficiency and quality. Additionally, while this study primarily focuses on cabbage, the techniques and methods proposed herein hold promise for application to crops with intricate root morphology, highlighting their broad applicability. Furthermore, this study offers a fresh perspective and methodology for future investigations in the realm of plant root pose detection, and carries substantial significance in terms of both technical innovation and practical application.

5.2. Limitations

While the method presented in this study has demonstrated promising experimental outcomes, there remains room for enhancement in both accuracy and generalization, particularly when dealing with small data sample sizes. The experimental results in Fig. 6 show that the accuracy and loss of the YOLOv5s model reach convergence after approximately 100 training iterations. However, the number of data samples in this study is relatively small, which may impact the results. Additionally, to visually assess the measurement bias between our method and actual measurements, an error bar chart was generated (Fig. 10). Although some cabbages exhibit relatively large absolute errors (up to 6.5°), contributing factors such as uneven lighting, root damage, and soil occlusion may have influenced these errors, as shown in Fig. 11 (a). The impact of these variables during image threshold processing could result in the binarized image lacking essential information, as shown in Fig. 11 (b), thereby leading to deviations between subsequent image processing and actual measurement results, as illustrated in Fig. 11 (c) and (d).

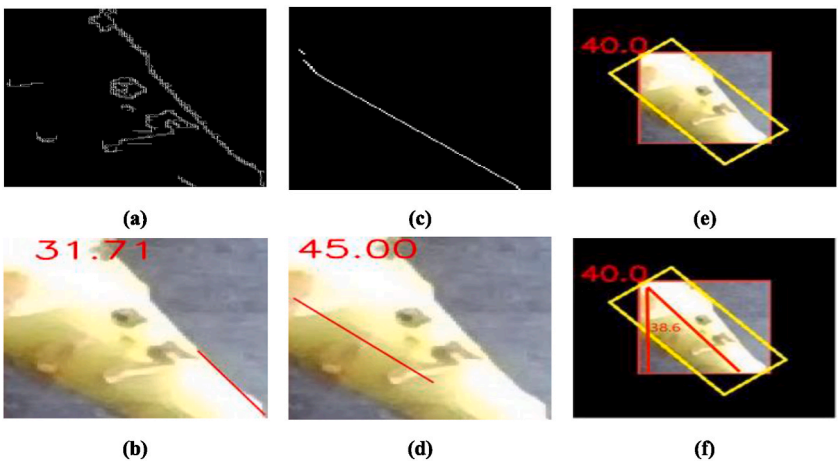


Fig. 9. Comparison of different methods for recognizing the posture of cabbage roots. (a). Canny method, (b). Hough transform line, (c). Skeleton extraction, (d). Fitting a straight line, (e). Minimum circumscribed rectangle, (f). Real measurement.

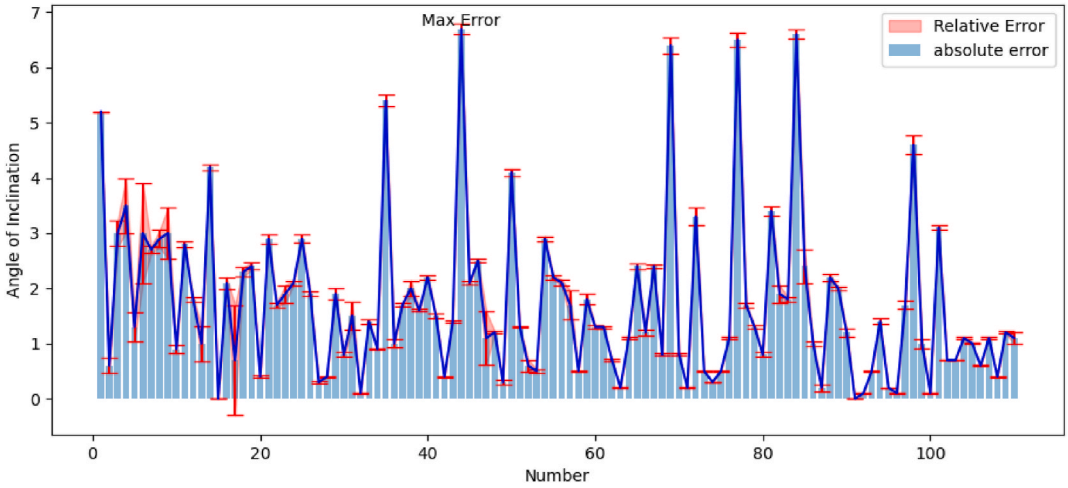


Fig. 10. Test error comparison between test value and actual value.

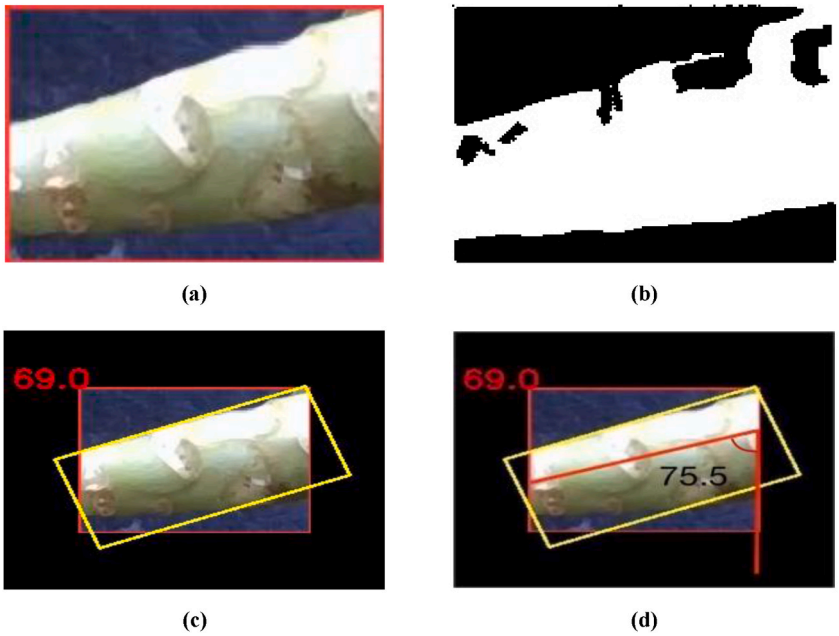


Fig. 11. Comparison results with actual measurement. (a). Uneven illumination, (b). Threshold processing, (c). Algorithm measurement results, (d). Actual measurement results.

5.3. Future work

The follow-up work of this study is expected to proceed on three fronts. Firstly, we plan to further acquire datasets of various cabbage varieties for training purposes, aiming to enhance the universality and robustness of the model. Secondly, in the prediction method of cabbage root tilt angle, in addition to the methods mentioned in this paper, we will further optimize it through techniques such as machine vision and fitting prediction with Bézier curves. Finally, the system will be integrated into a cabbage harvesting machine for practical cabbage harvesting assistance, thereby verifying the accuracy of pose recognition and root cutting.

6. Conclusion

This study proposes a cabbage posture recognition method that combines deep learning and traditional image processing techniques, demonstrating its ability to accurately identify cabbage postures. A root detection model based on the YOLOv5s network structure was used for training, and high-level features of the root were extracted through a deep convolutional network, successfully

overcoming the limitations of traditional image processing algorithms. Comparative analysis against principal detection models such as SSD, Fast R–CNN, ATSS, and RetinaNet revealed a substantial enhancement in precision and recall with the YOLOv5s model. The measurement based on the minimum circumscribed rectangle algorithm has a good correlation with the actual measurement. From the analysis of error indicator performance, the MAE, MSE, and RMSE of this study are 1.69 %, 4.90 %, and 2.21 respectively, and the MAPE is 15.13 %. This method reduces cabbage harvest-related damages, enhances operational efficiency, and decreases reliance on manual labor, with potential widespread applications in agricultural mechanization.

Fundings

This work was supported by Zhejiang Province Key Laboratory of Smart Management & Application of Modern Agricultural Resources under (Grant Number: 2020E10017).

Data availability statement

The data presented in this study are available on demand from the corresponding author 03120@zjhu.edu.cn.

CRediT authorship contribution statement

Fen Qiu: Writing – original draft, Investigation. **Chaofan Shao:** Software, Methodology, Investigation. **Cheng Zhou:** Writing – review & editing, Supervision, Funding acquisition. **Lili Yao:** Writing – review & editing, Methodology, Data curation, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] D. Du, J. Wang, L. Xie, F. Deng, Design and field test of a new compact self-propelled cabbage harvester, Transactions of the ASABE 62 (2019) 1243–1250.
[2] J. Zhang, G. Cao, Y. Jin, W. Tong, Y. Zhao, Z. Song, Parameter optimization and testing of a self-propelled combine cabbage harvester, Agriculture 12 (2022) 1610.
[3] L. Cao, S. Miao, Design of Chinese cabbage harvester, in: 2020 IEEE International Conference on Mechatronics and Automation (ICMA), IEEE, 2020, pp. 243–248.
[4] Q. Han, C. Chen, Research on tilt sensor technology, in: 2008 IEEE International Symposium on Knowledge Acquisition and Modeling Workshop, IEEE, 2008, pp. 786–789.
[5] S. Das, A simple, low cost optical tilt sensor, Int. J. Electron. Electr. Eng. 2 (2014) 235–241.
[6] A. Bunce, J.C. Volin, D.R. Miller, J. Parent, M. Rudnicki, Determinants of tree sway frequency in temperate deciduous forests of the northeast United States, Agric. For. Meteorol. 266 (2019) 87–96.
[7] Y. Park, H.I. Son, A sensor fusion-based cutting device attitude control to improve the accuracy of Korean cabbage harvesting, Journal of the ASABE 65 (2022) 1387–1396.
[8] K. James, C. Hallam, C. Spencer, Measuring tilt of tree structural root zones under static and wind loading, Agric. For. Meteorol. 168 (2013) 160–167.
[9] Q. Guo, R. Chen, X. Sun, M. Jiang, H. Sun, S. Wang, L. Ma, Y. Yang, J. Hu, A non-destructive and direction-insensitive method using a strain sensor and two single axis angle sensors for evaluating corn stalk lodging resistance, Sensors 18 (2018) 1852.
[10] S.M.M. Roomi, R.J. Priya, S. Bhumesh, P. Monisha, Classification of mangoes by object features and contour modeling, in: 2012 International Conference on Machine Vision and Image Processing (MVIP), IEEE, 2012, pp. 165–168.
[11] M. Hannan, T. Burks, D.M. Bulanon, A machine vision algorithm combining adaptive segmentation and shape analysis for orange fruit detection, Agric. Eng. Int.: CIGR J. XI (2009).
[12] X. Wei, K. Jia, J. Lan, Y. Li, Y. Zeng, C. Wang, Automatic method of fruit object extraction under complex agricultural background for vision system of fruit picking robot, Optik 125 (2014) 5684–5689.
[13] J. Rakun, D. Stajnko, D. Zazula, Detecting fruits in natural scenes by using spatial-frequency based texture analysis and multiview geometry, Comput. Electron. Agric. 76 (2011) 80–88.
[14] J. Wu, B. Zhang, J. Zhou, Y. Xiong, B. Gu, X. Yang, Automatic recognition of ripening tomatoes by combining multi-feature fusion with a bi-layer classification strategy for harvesting robots, Sensors 19 (2019) 612.
[15] J. Huixian, The analysis of plants image recognition based on deep learning and artificial neural network, IEEE Access 8 (2020) 68828–68841.
[16] Q. Zhang, Y. Liu, C. Gong, Y. Chen, H. Yu, Applications of deep learning for dense scenes analysis in agriculture: a review, Sensors 20 (2020) 1520.
[17] M. Momeny, A. Jahanbakhshi, K. Jafarnezhad, Y.-D. Zhang, Accurate classification of cherry fruit using deep cnn based on hybrid pooling approach, Postharvest Biol. Technol. 166 (2020) 111204.
[18] J. Yan, Y. Zhao, L. Zhang, X. Su, H. Liu, F. Zhang, W. Fan, L. He, Recognition of rosa roxbunghii in natural environment based on improved faster rcnn, Trans. Chin. Soc. Agric. Eng. 35 (2019) 143–150.
[19] Z. Wang, Y. Ling, X. Wang, D. Meng, L. Nie, G. An, X. Wang, An improved faster r-cnn model for multi-object tomato maturity detection in complex scenarios, Ecol. Inf. 72 (2022) 101886.
[20] P. Ardhianto, R.B.R. Subiakto, C.-Y. Lin, Y.-K. Jan, B.-Y. Liau, J.-Y. Tsai, V.B.H. Akbari, C.-W. Lung, A deep learning method for foot progression angle detection in plantar pressure images, Sensors 22 (2022) 2786.
[21] Y. Pusparani, C.-Y. Lin, Y.-K. Jan, F.-Y. Lin, B.-Y. Liau, P. Ardhianto, I. Farady, J.S.R. Alex, J. Aparajeeta, W.-H. Chao, et al., Diagnosis of alzheimer's disease using convolutional neural network with select slices by landmark on hippocampus in mri images, IEEE Access 11 (2023) 61688–61697.
[22] A.A. Prisilla, Y. Pusparani, W.-T. Chang, B.-Y. Liau, Y.-K. Jan, P. Ardhianto, C.-Y. Lin, C.-W. Lung, Automatic detection of lumbar disc herniation using yolov7, in: 2023 International Conference on Consumer Electronics-Taiwan (ICCE-Taiwan), IEEE, 2023, pp. 843–844.
[23] Y. Tian, G. Yang, Z. Wang, H. Wang, E. Li, Z. Liang, Apple detection during different growth stages in orchards using the improved yolo-v3 model, Comput. Electron. Agric. 157 (2019) 417–426.

[24] W. Ji, X. Gao, B. Xu, Y. Pan, Z. Zhang, D. Zhao, Apple target recognition method in complex environment based on improved yolov4, J. Food Process. Eng. 44 (2021) e13866.
[25] Z. Wang, L. Jin, S. Wang, H. Xu, Apple stem/calyx real-time recognition using yolo-v5 algorithm for fruit automatic loading system, Postharvest Biol. Technol. 185 (2022) 111808.
[26] Y. Xie, X. Zhong, J. Zhan, C. Wang, N. Liu, L. Li, P. Zhao, L. Li, G. Zhou, Eclpod: an extremely compressed lightweight model for pear object detection in smart agriculture, Agronomy 13 (2023) 1891.
[27] C. Dewi, O.M. Kamlasi, G. Chhabra, G. Dai, K. Kaushik, I.U. Khan, Automated fruit classification based on deep learning utilizing yolov8, in: 2023 10th IEEE Uttar Pradesh Section International Conference on Electrical, Electronics and Computer Engineering (UPCON), vol. 10, IEEE, 2023, pp. 801–807.
[28] R. Girshick, Fast r-cnn, in: Proceedings of the IEEE International Conference on Computer Vision, 2015, pp. 1440–1448.
[29] S. Ren, K. He, R. Girshick, J. Sun, Faster r-cnn: towards real-time object detection with region proposal networks, Adv. Neural Inf. Process. Syst. 28 (2015).
[30] J. Dai, Y. Li, K. He, J. Sun, R-fcn: object detection via region-based fully convolutional networks, Adv. Neural Inf. Process. Syst. 29 (2016).
[31] J. Redmon, S. Divvala, R. Girshick, A. Farhadi, You only look once: unified, real-time object detection, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 779–788.
[32] J. Redmon, A. Farhadi, Yolo9000: better, faster, stronger, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 7263–7271.
[33] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, A.C. Berg, Ssd: single shot multibox detector, in: Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I 14, Springer, 2016, pp. 21–37.
[34] M. Tan, R. Pang, Q.V. Le, Efficientdet: scalable and efficient object detection, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 10781–10790.
[35] L. Huipeng, L. Changyong, L. Guibin, C. Lixin, Picking point positioning of multi variety table grapes based on deep learning, Journal of Chinese Agricultural Mechanization 43 (2022) 155.
[36] X. Qi, J. Dong, Y. Lan, H. Zhu, Method for identifying litchi picking position based on yolov5 and pspnet, Rem. Sens. 14 (2022) 2004.