Key technologies of tomato-picking robots based on machine vision

Zirui Yin[1,2], Han Li[1,2*], Zhijiang Zuo[1,2], Zhaoxin Guan[1,2]

(1. *State Key Laboratory of Precision Blasting, Jianghan University, Wuhan 430056, China*;
2. *Hubei Key Laboratory of Blasting Engineering, Jianghan University, Wuhan 430056, China*)

Abstract: To address the challenges of harsh harvesting environments, high labor intensity, and low picking efficiency in tomato harvesting, this study investigates the key technologies related to the end-effector design, detection and recognition, and spatial localization of tomato-picking robots. A non-contact cavity-type end-effector is designed, which effectively prevents tomato damage caused by compression during picking while preserving the peduncle. Additionally, the motion of the robotic arm is simulated for performance analysis. Subsequently, tomato images are captured and annotated for training deep neural network models. Both the original YOLO v8n and the improved YOLO v8n models are used for tomato image detection, with a focus on the impact of varying light intensities and different tomato maturities on recognition and localization accuracy. Experimental results demonstrate that the robot's vision system achieves optimal recognition and localization performance under light intensities ranging from 20 000 to 30 000 lx, with an accuracy of 91.5%, an average image detection speed of 15.1 ms per image, and an absolute localization error of 1.55 cm. Furthermore, the prototype tomato-picking robot's end-effector successfully performed stable grasping of individual tomatoes without damaging the skin, achieving a picking success rate of 83.3%, with an average picking time of approximately 9.5 s per fruit. This study provides a technical support for the automated harvesting of tomato-picking robots.

Keywords: tomato-picking robot, end effector, machine vision, deep learning, YOLO v8n
DOI: 10.25165/j.ijabe.20251803.8954

Citation: Yin Z R, Li H, Zuo Z J, Guan Z X. Key technologies of tomato-picking robots based on machine vision. Int J Agric & Biol Eng, 2025; 18(3): 247–256.

1 Introduction

Tomatoes are widely appreciated for their unique flavor and nutritional value, offering significant economic potential. As a major contributor to global tomato production, China has become the largest supplier of raw materials for tomato processing products, with an annual output exceeding one-third of the global total. The industry continues to expand in scale[1,2]. To meet growing market demand, the proportion of greenhouse-grown tomatoes in China is steadily increasing, resulting in rising production volumes[3,4]. In tomato cultivation, the picking stage accounts for 40% to 50% of the labor input[5]. However, manual picking remains the predominant method, which poses challenges such as harsh working environments, low picking efficiency, and high labor costs[6,7]. With technological advancements and increasing labor shortages, picking robots have been introduced in agricultural production, improving labor efficiency and reducing economic losses caused by delayed picking[8-10]. However, the main technological barriers to the development of picking robots lie in target recognition, localization, and fruit separation. Therefore, the design and research of the tomato-picking robot provide theoretical support for the automation of agricultural production equipment in China's facility agriculture and contribute to the development of tomato-picking robots.

In recent years, numerous studies on tomato-picking robots have been conducted worldwide, but the picking performance still requires improvement. In the early 1990s, Japan developed a tomato-picking robot[11], with a soft-pad end-effector design to reduce fruit damage. However, the picking process required manual assistance, achieving a success rate of approximately 70%. At Jiangsu University, Liu et al. developed a mobile tomato-picking robot[12,13] equipped with a wheeled chassis, a clamping and cutting end-effector, and a stereo vision system. This robot aimed to collaborate with greenhouse fruit-picking and transport robots to achieve full automation, including tomato picking, on-site grading, collection, transportation, and unloading. Yaguchi et al. designed a large-scale tomato-picking robot[14] with a rotating picking gripper. The average picking time per fruit was about 23 s, with a success rate of only 60%, and the process was easily disrupted. Wang et al. developed a tomato-picking robot with a sleeve-and-airbag-based end-effector[15], capable of picking tomatoes within 24 s per fruit, though the harvested fruit lacked peduncles. Zheng et al. proposed a nested approach to tomato picking[16], simplifying the process and reducing damage. The improved end-effector achieved a 57.5% success rate within 14.9 s per fruit. Rong et al. designed an integrated adsorption-gripping robotic hand[17], optimizing picking strategies to significantly reduce the impact of collisions on grasping. The tomato picking success rate increased to 72.1%, with an average time of 14.6 s per fruit. Fujinaga et al. developed a suction-based cutting device for tomato picking[18], achieving a 52.4% success rate. However, obstacles surrounding the fruit presented significant challenges. Overall, current tomato-picking robots face limitations in fruit protection, picking efficiency, and methodology. During the picking process, tomatoes are prone to mechanical damage, such as compression and scratches, particularly for highly mature or thin-

Received date: 2024-03-24 **Accepted date:** 2025-03-01

Biographies: **Zirui Yin**, Graduate students, research interest: machine vision, Email: 2177105922@qq.com; **Zhijiang Zuo**, Professor, research interest: machine vision, Email: zzjdfcy@163.com; **Zhaoxin Guan**, Graduate students, research interest: visual control, route planning, Email: 929691569@qq.com.

***Corresponding author:** **Han Li**, Lecturer, research interest: visual sevoing control. College of Intelligent Manufacturing, Jianghan University, P.O. Box 47, Jianghan University, No. 8, Sanjiaohu Rd., Wuhan Economic & Technological Development Zone, Hubei, 430056, China. Tel: +86-27-84226912, Email: gracelihan@jhun.edu.cn.

skinned tomatoes. Existing end-effector designs fail to entirely mitigate these issues. Furthermore, many robots cannot effectively preserve peduncles, negatively impacting the fruit's appearance and subsequent storage quality.

Although tomato-picking robots have not yet been commercialized, related detection algorithms have undergone extensive research. Early fruit recognition methods primarily relied on digital image processing and machine learning techniques[19-22], which depended on manually designed features. For instance, Benavides et al. used color segmentation methods to determine tomato picking points[23], but the threshold settings required agricultural expertise and were inadequate under complex lighting conditions. Traditional fruit detection methods faced challenges such as semantic information extraction in complex backgrounds, occlusions, and uneven lighting[24-26]. Recently, deep learning-based fruit recognition methods have become mainstream[27]. Convolutional neural networks can automatically learn features from training data, demonstrating superior recognition performance in complex scenarios. Among deep learning models, the YOLO series has gained attention for its high accuracy and fast detection speed[28]. Li et al.[29] proposed a YOLO v4+HSV (Hue, Saturation, Value) method, achieving a recognition accuracy of 94.77% for mature tomatoes with a specific proportion of 16% in the test set. Wang et al.[30] introduced an improved SM-YOLOv5 detection algorithm, enhancing recognition precision in greenhouse environments to 97.8% with a model size of just 6.33 MB. Cai et al.[31] utilized multimodal RGB-D perception and an improved YOLOv7-tiny network for cherry tomato detection, improving real-time detection accuracy and precision over existing methods. Miao et al.[32] proposed a lightweight YOLO v7 model for cherry tomato maturity detection, achieving precision, recall, and mean average precision rates of 98.6%, 98.1%, and 98.2%, respectively, with a model memory footprint of 66.5 MB. As an advanced object detection model, YOLO v8 offers significant improvements in computational efficiency and detection accuracy, making it well-suited for tomato detection tasks.

To address the limitations of existing tomato-picking robots in terms of end-effector design, picking efficiency, and visual detection, this study presents the design of a tomato-picking robot for greenhouse applications. A non-contact end-effector is developed to reduce direct contact with tomatoes, effectively minimizing damage during the picking process while successfully preserving the fruit's peduncle. Additionally, an improved YOLO v8n-based vision system is proposed for tomato detection, and the image recognition performance of both the original YOLO v8n and the improved YOLO v8n models under different lighting conditions is compared. Finally, a series of experiments are conducted to evaluate the robot's visual system performance in terms of detection accuracy, localization precision, and picking efficiency.

2 Materials and methods

2.1 Tomato experimental data

2.1.1 Tomato dataset creation

To minimize the impact of scene variables, an experimental field measuring 1.20 m×3.00 m×0.35 m was established near the laboratory. An integrated meteorological multi-element sensor (Shandong Renke Measurement and Control Technology Co., Ltd., Jinan, China) was used to measure light intensity. Data collection was conducted during the tomato ripening period from May to June, divided into three time slots each day: 9:00-11:00 AM, 1:00-3:00 PM, and 5:00-7:00 PM. The light intensity during image capture was classified into four levels based on natural lighting conditions: above 30 000 lx, 20 000-30 000 lx, 1000-20 000 lx, and 0-1000 lx. When the natural light intensity was below 1000 lx, supplementary lighting equipment was used to provide a light intensity of 15 000-20 000 lx. Images were captured from both front-lit and backlit angles, with a fixed shooting distance of 50 cm to ensure sufficient detail and comprehensive coverage. Additionally, to further enrich the dataset, supplementary data were collected from the tomato cultivation area at the Agricultural Institute of Wuhan, Hubei Province.

After excluding invalid samples from the raw images, 1000 images containing tomatoes at varying maturity levels were selected as the initial dataset. Data augmentation techniques, such as mirroring, cropping, rotation, and the addition of Gaussian noise were applied to expand the dataset, as shown in Figure 1. Ultimately, 3500 valid images were obtained, with 80% used for training, 10% for validation, and 10% for testing.

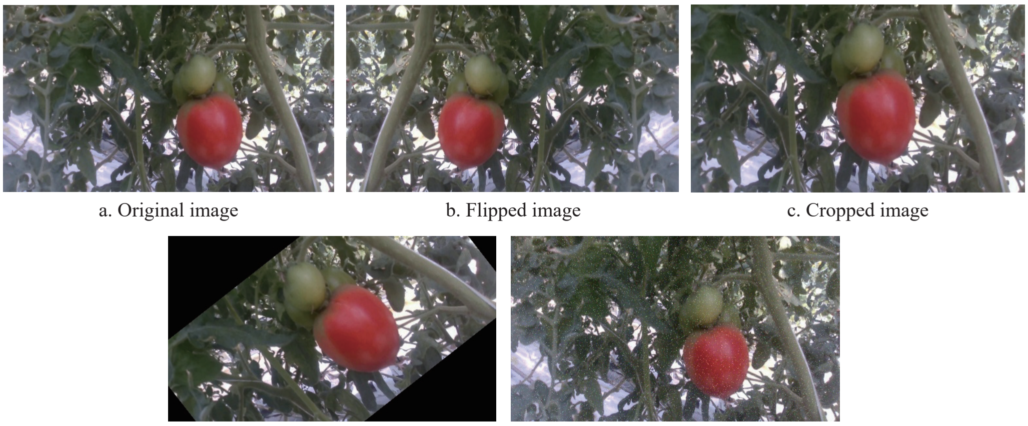
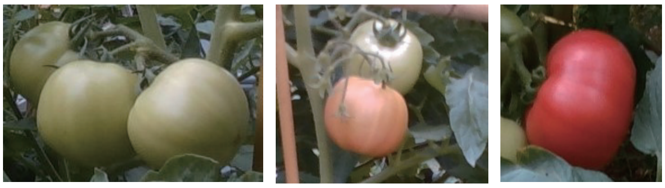

a. Original image b. Flipped image c. Cropped image
d. Rotated image e. Image with added Gaussian noise

Figure 1 Tomato dataset augmentation results

2.1.2 Classification of tomato maturity levels

The target objects of the tomato-picking robot are fruits that meet the conditions for picking, necessitating the classification of fruit maturity levels. According to the "National Standards of the People's Republic of China-Tomato"[33,34], tomatoes in the immature and green ripening stages are categorized as raw tomatoes; those in the turning stage, early red ripening stage, and mid-red ripening stage are categorized as transitional tomatoes; while those in the late red ripening and overripe stages are categorized as mature tomatoes. The specific growth stages of tomatoes are shown in Figure 2. The labelme tool was then used to annotate tomatoes in the images, and their maturity levels were classified based on this standard.



a. Raw stage b. Transition stage c. Mature stage

Figure 2 Tomato growth stages

2.2 Tomato-picking robot structural design

2.2.1 Prototype construction

An experimental platform for the tomato-picking robot was built. The robot is equipped with two differential motors, powered by electricity, with performance capabilities that meet the mobility requirements in a greenhouse environment. The chassis is designed as a box structure, housing the main control computer and communication equipment, with sealed gaps to meet the operational needs of high temperature and high humidity in the greenhouse. The prototype of the tomato-picking robot is shown in Figure 3. The robot mainly consists of the following key components: 1) UR3 robotic arm, responsible for precise operations and fruit picking; 2) Vision system, used for real-time tomato detection and providing positioning information; 3) End effector, responsible for cutting the tomato peduncle; 4) Mobile chassis system, ensuring autonomous movement of the robot within the greenhouse; 5) Main control unit, acting as the control center of the entire system, coordinating the operation of each module.

2.2.2 End effector design

Considering the large size and delicate, scratch-sensitive skin of tomatoes, a cavity-type constraining method was selected, which fixes the tomato peduncle for forceful shearing. The process for cutting the peduncle with the designed end effector is as follows:

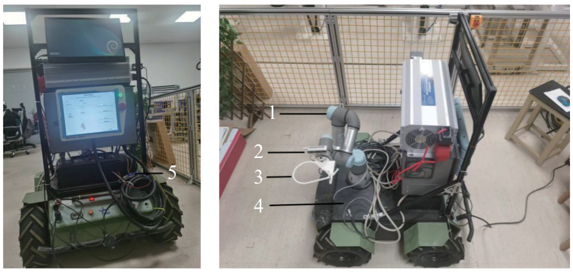


Figure 3 Overall structure of the tomato-picking robot

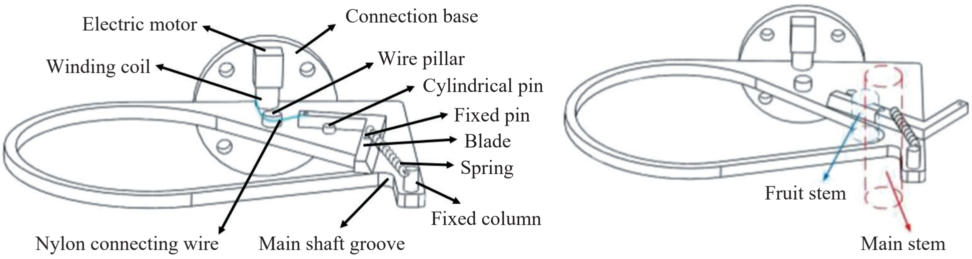
1) Utilizing the space beneath the tomato fruit, the target is looped from below, and the end effector halts horizontally above the peduncle;

2) The end effector is then moved horizontally, placing the peduncle into the gradually narrowing cutting slot;

3) The motor begins to rotate, pulling the connecting wire around the coil, and the L-shaped blade starts rotating to sever the peduncle in the cutting slot;

4) After cutting, the motor reverses, and the spring retracts to return the blade to its original position.

The end effector, as shown in Figure 4, is designed with an outer fixed frame that isolates leaves and non-target fruits to resist interference. The clamping cutting slot prevents the fruit peduncle from slipping out of the cutting position. The L-shaped blade is embedded to prevent fruit damage during movement, and its cutting edge is parallel to the horizontal plane of the end effector, providing greater effective cutting force. The guide post provides a consistent and stable pulling force in the same direction for the force end of the L-shaped blade. The stopper pin ensures that the opening and closing motion of the L-shaped blade during operation stays within the effective return range.



a. End effector in preparation state b. End effector in cutting state

Figure 4 The end effector

2.2.3 Robotic arm simulation motion analysis

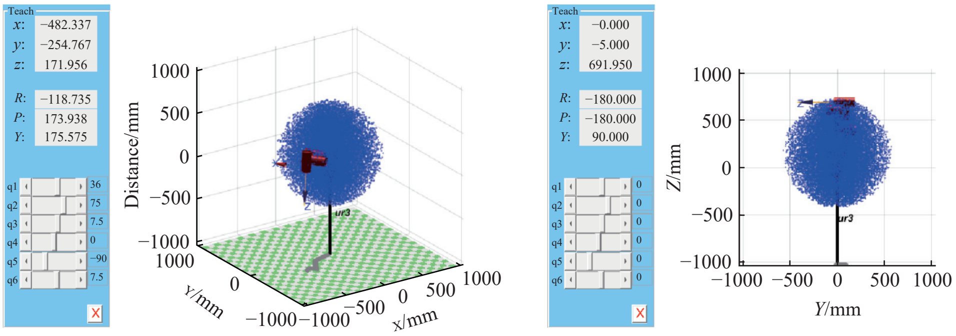
The picking experiment uses the UR3 robotic arm (Universal Robots A/S, Odense, Denmark), which is widely used and manufactured by Universal Robots. The robotic arm features six degrees of freedom, with a flexible and stable design that can sensitively detect changes in motor torque. In the event of a collision or obstruction, the arm automatically triggers a self-locking mechanism, halting movement promptly to ensure the safety of the robot, greenhouse facilities, and operators. The robotic arm weighs only 11.2 kg and can carry an effective load of up to 3 kg, with motion precision down to the millimeter level, making it well-suited for the precise movements required in tomato harvesting. The detailed performance parameters of the robotic arm are listed in Table 1.

To validate the motion behavior of the end-effector of the UR3 robotic arm within its workspace and evaluate whether it meets the operational requirements, this study employs MATLAB for simulation analysis. By conducting a workspace analysis of the robotic arm fixed at a certain point in the spatial coordinate system, as shown in Figure 5, it is determined that the UR3 robotic arm's

Table 1 Performance parameters of the UR3 robotic arm

Serial number	Item	Parameter
1	Effective payload	3 kg
2	Working range	Spherical space with a 500 mm radius
3	Joint range	+/–360°
4	Speed	Wrist joint: 180°/s; TCP: 1 m/s
5	Repeatability	+/–0.03 mm with effective payload
6	I/O power	Control box: 24 V 2 A; Tool side: 12 V/24 V, 600 mA short-term 2 A
7	Communication	Control frequency: 500 Hz; ModbusTCP: signal frequency 500 Hz

workspace is distributed in a spherical shape with a radius of 500 mm. The density of points within the workspace reflects the probability that the arm's end-effector can reach those points. The region closer to the robotic arm's body has a higher density of points, indicating that the arm's mobility is more flexible in this area, with more potential solutions for movement paths.



a. Axial side view

b. *Y-Z* direction view

Figure 5 Workspace of the UR3 robotic arm

Two points (–100,–100,–100) and (260,360,160) within the workspace were selected for motion planning research. Five-point function interpolation was performed for joint space motion planning, and then Cartesian space motion planning was carried out using the 'ctraj', 'jtraj', and 'trinterp' functions. The two motion trajectories are shown in Figure 6. Analysis of the trajectories shows that motion planning in joint space involves identical steps for each joint, indicating synchronized driving. This means that each joint consumes the same amount of time in its respective path to ensure smooth motion. However, this planning method makes it difficult to effectively control the position of the end effector in real time, resulting in a large sweeping motion that does not meet the application requirements of tomato-picking robots in greenhouse facilities. The analysis results of the end effector's motion state are visualized. In contrast, using Cartesian space motion planning ensures the constraint of the end effector's position change. The analysis of its spatial position over time is shown in Figure 7.

Based on the generated graphs, the UR3 robotic arm performs trapezoidal interpolation for speed in the Cartesian coordinate system, ensuring that the position of the end effector changes over time. The motion trajectory in three-dimensional space forms a straight line, and the speed curve exhibits a trapezoidal shape, with smooth transitions at the corners, indicating a steady change between acceleration, constant speed, and deceleration states. After startup, the angular change curves of certain joints show noticeable inflection points, likely occurring near singularities in the motion trajectory. However, overall, the analysis indicates that the motion

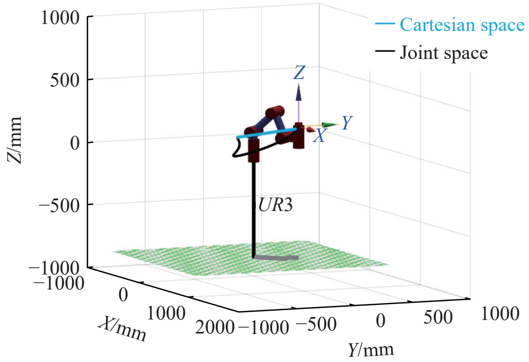
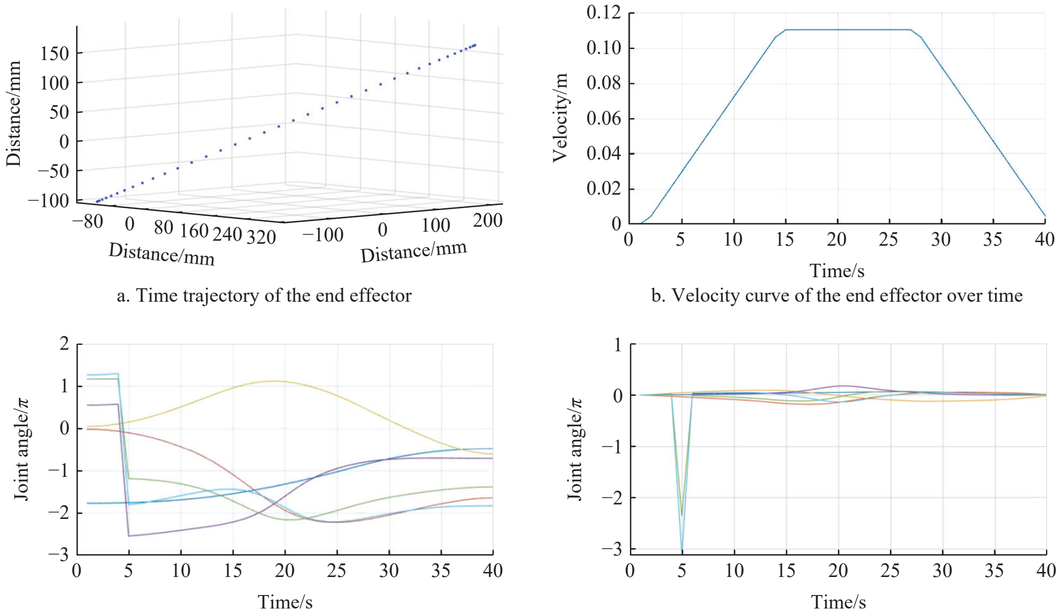


Figure 6 Two trajectories planned using two different planning methods



a. Time trajectory of the end effector

b. Velocity curve of the end effector over time

c. Joint angles from initial to target position over time

d. Joint angle changes over time

Figure 7 Cartesian space motion planning analysis

path after the end effector is mounted meets the required operational path.

2.3 Tomato fruit detection and localization

2.3.1 Detection

YOLO v8, as a one-stage object detection algorithm, offers advantages in both speed and accuracy. It can learn the color, shape, and texture features of objects by combining feature maps of different scales, which helps achieve high precision in close-range object detection tasks. Considering the intelligent recognition application on agricultural machinery, emphasis is placed on the real-time detection capability and the lightweight nature of the model. Therefore, the YOLO v8n network model is chosen. YOLO v8n is the smallest model in the YOLO v8 series, featuring a smaller network depth and feature map size. It achieves faster detection speed while maintaining detection accuracy, with lower model memory consumption, making it suitable for lightweight, high-precision, and real-time tomato fruit detection. The YOLO v8n model consists of four main parts: Input, Backbone, Neck, and Head. The Input part is responsible for receiving raw image data and preprocessing it to provide suitable input for subsequent feature extraction and object detection. The Backbone part extracts multi-level semantic features from the input image using convolutional layers and other feature extraction modules. The Neck part uses a feature fusion module to effectively combine feature maps from different scales, enabling more precise localization and identification of objects at multiple scales. The Head part is primarily responsible for classifying the objects and predicting the bounding box coordinates of the objects.

To ensure the performance of the original YOLO v8n model in object detection while reducing network model parameters and improving the detection accuracy for tomato fruits, this study introduces the following improvements to the original YOLO v8n model. The entire Backbone network in the original model is replaced with the ShuffleNetV2 structure, which reduces the number of parameters during network deployment and training, thus achieving a lightweight network design. Additionally, the CBAM (Convolutional Block Attention Module) attention mechanism is introduced into the Neck network detection head to reduce network complexity while ensuring the accuracy of tomato fruit detection. The improved YOLO v8n network structure is shown in Figure 8.

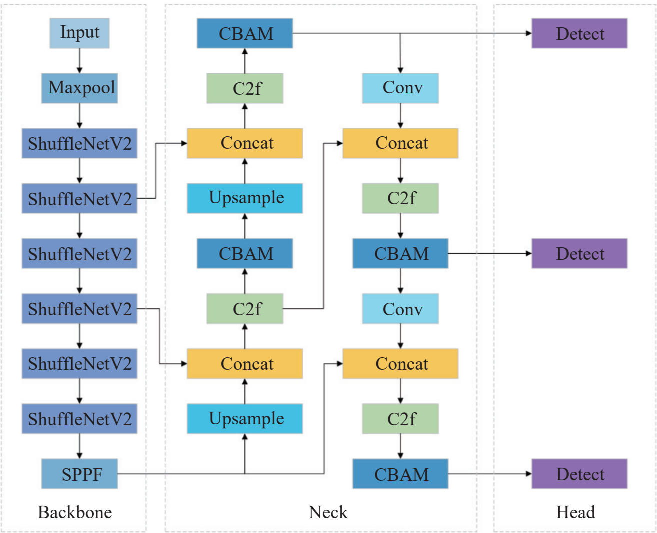


Figure 8 The improved YOLO v8n network architecture

2.3.2 Localization

Based on the camera imaging model, stereo vision involves the pixel coordinate system, image coordinate system, camera coordinate system, and world coordinate system, as shown in Figure 9. O_w-$X_wY_wZ_w$ describes the position of the camera in the world coordinate system; O_c-$X_cY_cZ_c$ represents the coordinate system of the Realsense D435 camera installation, with the optical center as the origin; O-XY refers to the tomato image coordinate system, with the optical center at the center of the image; O_o-uv is the pixel coordinate system, with the origin at the top-left corner; P_w is a point on the tomato, with its coordinates in the image coordinate system denoted as P_d. The focal length of the camera is the distance between O and O_c, f=‖O-O_c‖.of the camera is the distance between O and O_c, f=‖O-O_c‖.

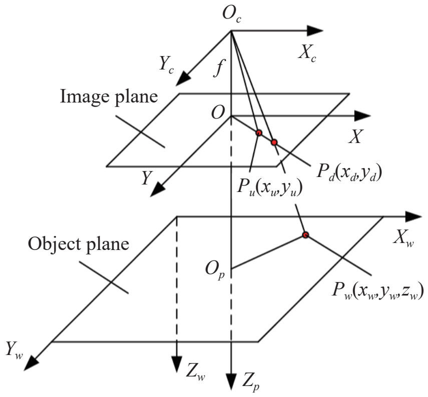


Figure 9 Camera model coordinate system transformation model

Through the transformation model relationship in Figure 9, the position of the tomato fruit in the world coordinate system is converted to the pixel point position in the pixel coordinate system, as shown in Formula 1.

$$Z_c\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} \frac{1}{dx} & 0 & u_0 \\ 0 & \frac{1}{dx} & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} R & t \\ 0^T & 1 \end{bmatrix} \begin{bmatrix} X_W \\ Y_W \\ Z_W \\ 1 \end{bmatrix} = \begin{bmatrix} f_x & 0 & u_0 & 0 \\ 0 & f_y & v_0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} R & t \\ 0^T & 1 \end{bmatrix} \begin{bmatrix} X_W \\ Y_W \\ Z_W \\ 1 \end{bmatrix} \tag{1}$$

In the formula, $\begin{bmatrix} f_x & 0 & u_0 & 0 \\ 0 & f_y & v_0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$ denotes the camera intrinsic parameters, while $\begin{bmatrix} R & t \\ 0^T & 1 \end{bmatrix}$ denotes the camera extrinsic parameters.

The above model can only describe the ideal geometric relationship. However, in practical applications, there are often assembly errors during the installation and production of the camera, which lead to optical distortion and image distortion. To address this issue, it is necessary to calibrate and correct the camera to eliminate the distortion and recompute accurate spatial coordinates. In the calibration process, the Realsense D435 camera is used, which provides two calibration methods: rectification calibration and depth scale calibration. The depth scale calibration method is equipped with a graphical user interface (GUI) that simplifies the calibration process and enhances operational convenience. During the calibration process, the calibration board is kept at an appropriate distance from the camera to ensure the board is fully displayed within the camera's field of view. When the camera in calibration mode detects the calibration board, it will display corresponding guidance prompts through the GUI. The board is then moved until the blue area in the camera image disappears. At this point, the camera automatically performs the scale calibration, and the RGB image is also recalibrated, as shown in Figure 10. If both calibration steps are successfully completed, the system will prompt to exit the guidance process. If the calibration fails, the system will re-enter the guidance prompts for adjustments. If the expected accuracy is not achieved, the calibration process will be repeated.

The camera's intrinsic and extrinsic parameters were obtained through calibration and then used in the calculation as shown in Formula 1. The extrinsic parameter matrix of the camera is:

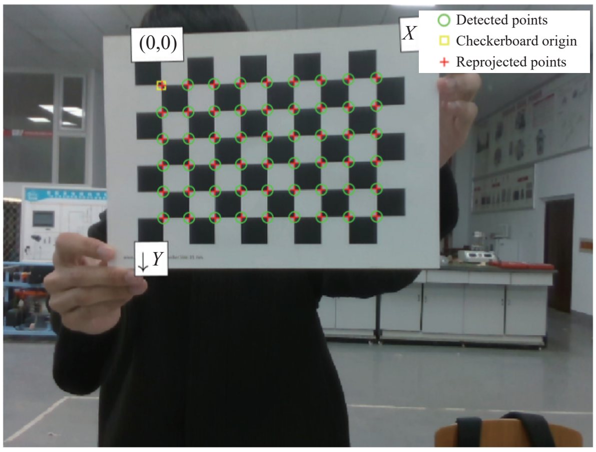


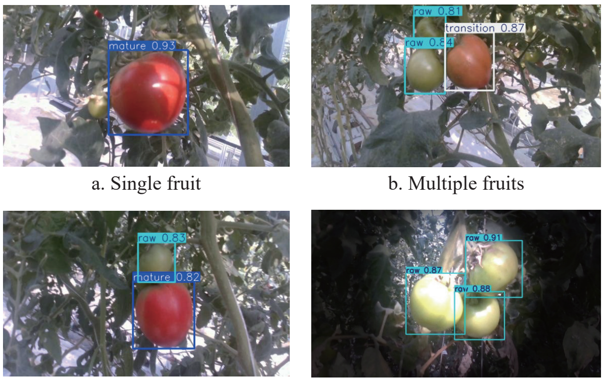
Figure 10 Camera calibration through calibration board

$$\begin{bmatrix} 0.999\,794\,492\,5 & -0.020\,103\,978\,6 & 0.002\,608\,214\,238 & -0.033\,137\,568\,29 \\ 0.020\,096\,246\,71 & 0.999\,793\,675\,6 & 0.002\,957\,532\,592 & -0.095\,621\,702\,14 \\ -0.002\,667\,134\,272 & -0.002\,904\,509\,48 & 0.999\,992\,225\,1 & 0.046\,786\,798\,83 \\ 0 & 0 & 0 & 1 \end{bmatrix}。$$

3 Results and discussion

3.1 Tomato fruit detection performance

The experimental setup for tomato target detection in this study is as follows: the operating system is Ubuntu 16.04, and the neural network training is conducted in the Anaconda3 virtual environment using Python 3.9. For hardware acceleration, the GPU parallel computing architecture utilizes CUDA 11.7, and the deep neural network acceleration library is cuDNN 11.2. During network training, the image resolution is set to 640×640, the initial learning rate is set to 0.01, the batch size is 16, the momentum is 0.937, the weight decay is 0.0005, and the number of training epochs is 200. The optimizer used is Stochastic Gradient Descent (SGD). Some of the detection results are shown in Figure 11. Overall, the model demonstrates good adaptability to different scenarios, with high accuracy in target detection.



a. Single fruit

b. Multiple fruits

c. Sufficient lighting

d. Under supplementary lighting

Figure 11 Tomato fruit detection cases

To evaluate the impact of different lighting conditions on tomato detection performance, both the improved YOLO v8n network and the original YOLO v8n network were used to test tomato images under five different lighting conditions in the test set. The results showed that the average recognition accuracy for the four natural light intensity images using the improved YOLO v8n network was 84.3%, 88.3%, 85%, and 81.6%, with an average recognition accuracy of 84.3% under supplementary lighting conditions, and a detection speed of 15.1 ms per image. In contrast, the original YOLO v8n network achieved an average recognition accuracy of 82.3%, 85.3%, 83%, and 80.6%, with an average recognition accuracy of 83.3% under supplementary lighting conditions, and a detection speed of 18.3 ms per image. The detection results were recorded and averaged, and the statistics are shown in Table 2. The experimental results indicate that the improved YOLO v8n network performed best in tomato image recognition under lighting conditions of 20 000-30 000 lx, showing stronger robustness, and the improved YOLO v8n network also exhibited faster detection speed.

Table 2 Detection results of the two detection models under different light intensities

Light intensity level	Recognition targets	Proposed model/%	YOLO v8n /%
30 000-	Raw	84	83
	Transition	82	79
	Mature	87	85
20 000-30 000	Raw	88	85
	Transition	85	82
	Mature	92	89
1000-20 000	Raw	85	83
	Transition	84	81
	Mature	86	85
0-1000	Raw	81	80
	Transition	80	78
	Mature	84	84
Supplementary lighting equipment	Raw	83	83
	Transition	84	82
	Mature	86	85
Detection speed (ms/pic)	-	15.1	18.3

In order to further verify the effectiveness of the proposed method in tomato detection, the improved YOLO v8n model is compared with YOLO v3-tiny, YOLO v5n, YOLO v6n, YOLO v7-tiny, and YOLO v8n lightweight object detection models. The performance of each model is evaluated using comprehensive metrics, including Precision (P), Recall (R), Mean Average Precision (mAP), Floating Point Operations (FLOPs), the number of parameters (Param), and model memory usage. The validation set is used for testing. Precision reflects the accuracy of the model's predictions, i.e., the proportion of true positive samples among all predicted positive samples. Recall reflects the completeness of the model's predictions, i.e., the proportion of true positive samples correctly predicted by the model. Mean Average Precision considers both precision and recall and provides a comprehensive evaluation of the detection algorithm's performance. FLOPs reflect the model's computational complexity, indicating the number of floating point operations required per second. The number of parameters is a measure of the model size, with fewer parameters typically indicating a smaller model and lower computational cost. Model memory usage reflects the memory requirements of the model during deployment, with lower memory usage facilitating efficient operation on resource-constrained devices.

The experimental results of different networks are listed in Table 3. Compared with other algorithms, the improved YOLO v8n model exhibits reductions in computational cost and parameter count to varying degrees, while demonstrating superior performance in terms of P, R, and mAP@0.5, reaching 94.1%, 89%, and 94.8%, respectively. Compared to the original YOLO v8n, the improved YOLO v8n model achieves a 34.3% reduction in model size by introducing lightweight convolutional modules, thus improving computational efficiency. Overall, the experimental results show that the improved YOLO v8n algorithm performs excellently and is particularly suitable for real-time tomato detection. This algorithm not only achieves higher detection efficiency, but also features a smaller model size, lower computational cost, and higher detection accuracy, making it highly suitable for deployment on terminal devices.

Table 3 Test results of different models

Model name	P/%	R/%	mAP@50/%	GFLOPs	Param/M	Model size/MB
Yolo v3-tiny	89.6	88.6	93.4	18.9	12.12	23.20
Yolo v5n	93.1	86.7	94.5	7.2	2.50	5.02
Yolo v6n	91.3	89.4	94.5	11.8	4.23	8.28
Yolo v7-tiny	91.1	85.3	92.8	13.2	6.02	11.60
Yolo v8n	93.7	88.2	93.6	8.2	3.01	5.94
Proposed model	94.1	89.0	94.8	5.2	1.93	3.90

3.2 Localization performance

Based on the prediction boxes detected by the improved YOLO v8n model and the data from the D435 depth camera, the pixel values corresponding to the depth distances are extracted and converted into Cartesian coordinates. The specific steps are as follows:

1) Using the trained deep learning object detection model, image features are extracted from the RGB image to detect the positions of the tomatoes in the image. The boundary box of the tomato targets in the image is calculated to obtain the coordinates (cx_i,cy_i,w_i,h_i), where cx_i and cy_i are the X and Y coordinates of the boundary box center, and w_i and h_i are the width and height of the boundary box, respectively.

2) Perform color filtering on the detected tomatoes to confirm the mature fruits that need to be picked and record the center coordinates of the selected fruits.

3) Based on the center pixel coordinates of the boundary box of the target tomato fruit, extract the corresponding spatial point depth value from the depth data, as shown in Formula 2:

$$\text{depth} = \text{scale} \times \text{Depth} \times [cx_i] \times [cy_i] \tag{2}$$

The scale represents the correction factor for the scale.

The center pixel points (cx_i,cy_i) and depth are substituted into Formula 2 to calculate the spatial coordinates of the tomato fruit (cx'_i,cy'_i,cz'_i) in the world coordinate system. After integrating the positioning algorithm into the ROS system, field tests are conducted under different lighting levels. The test process and results are shown in Figure 12. The distance between the tomato and the depth camera is fixed at 50 cm as the reference value for the experiment. Based on the principle of stereo vision ranging, the distance from the tomato to the depth camera is obtained by calculation. Each set of lighting conditions is repeated 10 times, and the average value of the measurement results is taken as the final data. At the same time, the measurement results are compared with the actual distance measured manually, and key indicators such as absolute error and relative error are calculated to evaluate the distance measurement performance of the algorithm under different lighting conditions. The experimental results are listed in Table 4.

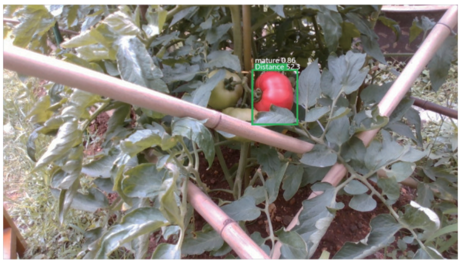


Figure 12 Distance detection results integrated into ROS

Table 4 Comparison between localization experimental data and real data

Light intensity level	Tomato-to- camera actual distance/cm	Measured average value/cm	Absolute error/cm	Relative error/%
≥30 000	50	52.10	2.10	4.2
20 000-30 000	50	51.55	1.55	3.1
1000-20 000	50	51.70	1.70	3.4
0-1000	50	48.65	1.65	3.3
Supplementary lighting equipment	50	48.05	1.95	3.9

The data in the table indicate that different light intensities interfere with depth distance measurements. The measurement accuracy is optimal when the light intensity is between 20 000-30 000 lx, as the lighting is sufficient but not overly strong, providing the best performance of the depth camera. When the light intensity exceeds 30 000 lx, issues such as overexposure cause the absolute error to increase to 2.1 cm. When the light intensity is between 1000-20 000 lx, the measurement accuracy is slightly lower than under optimal lighting conditions. In low-light conditions (0-1000 lx), although the error is relatively small, the performance of the depth camera is limited. With supplemental lighting, the performance improves compared to natural light conditions.

3.3 Picking performance

Field picking experiments were conducted in a trellis-based cultivation experimental field where the tomato plants were in good condition, and obstacles around the experimental area were cleared. The experiment took place on a sunny day with an average temperature of 24°C and no wind, as shown in Figure 13. Real-time monitoring of the light intensity in the experimental field was conducted using sensors to ensure the light intensity was between 20 000-30 000 lx. The tomato detection results in the experimental field are shown in Figure 10, where the YOLO model detected mature tomatoes with confidence levels of 89% and 49%, with distances of 32.7 cm and 37.3 cm, respectively. Additionally, the number of tomatoes observed by the tomato- picking robot's vision system and the number of tomatoes detected and identified are summarized in Table 5.

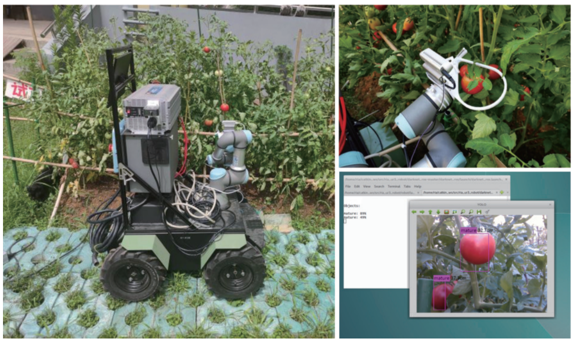


Figure 13 Tomato-picking robot testing

Table 5 Statistics of detected and identified fruit count

Tomato type	Number of observable tomatoes	Number of tomato detections	Number of false detections	Number of missed detections
Mature	72	68	0	4
Transitional	36	29	2	5
Raw	45	41	1	3
Total	153	138	3	12

The data in Table 5 show that the tomato recognition accuracy reaches 91.5%. Among them, mature tomatoes have the best recognition accuracy. However, there are two misclassifications for the transitional tomatoes; specifically, raw tomatoes were misclassified as transitional tomatoes. This occurred because the color and texture features of the transitional tomatoes are similar to those of the raw tomatoes, leading to some confusion during the recognition process. One misclassification occurred for raw tomatoes, mainly due to the misidentification of green leaves as raw tomatoes. This was due to the similarity in color and shape features between the leaves and raw tomatoes, and the complex background increased the difficulty of detection. Tomatoes that were partially occluded by leaves were still detected, but the confidence decreased significantly as the area of occlusion increased. Tomatoes with a larger area of occlusion and a more complex position not only introduced high measurement deviations in distance values, but also hindered the subsequent picking process of the robot. Therefore, setting the picking target as tomato with a confidence above 0.7 helps to plan better motion trajectories, reduce leaf interference, thus decreasing the robot's operating time while improving the picking success rate.

In the field picking tests, the robot's motion mode was set to trajectory planning mode. During the testing period, 72 mature tomatoes were picked across five experimental trials. The experimental results are listed in Table 6, where 60 tomatoes were successfully picked, resulting in a success rate of 83.3%. The average picking time per tomato was 9.5 s, and the robotic arm operated smoothly throughout the picking process. The harvested fruits exhibited no damage and retained their peduncle, as shown in Figure 14, without causing any harm to the planting environment. In failed attempts, 6.9% of the failures were due to the end effector being blocked by the tomato and unable to reach the target position, and 9.7% of the failures were due to the tomato's peduncle being hidden, leading to fruit damage during peduncle cutting. During the experiment, the robot's arm speed was limited to ensure smooth operation. The results validated the feasibility of the designed end effector during picking, reducing the mechanical contact pressure on the tomato skin and increasing the tolerance for positioning when cutting the peduncle, thus improving the tomato picking efficiency.

Table 6 Picking test records

Group number	Average diameter/ mm	Number of picks required	Number of targets reached	Number of peduncles retained	Number of picking successes	Average time/s
First	63	13	12	12	11	8.3
Second	77	18	16	15	14	10.7
Third	72	15	15	14	13	9.5
Fourth	89	14	14	13	12	10.6
Fifth	85	12	10	10	10	8.6
Total	77.2	72	67	65	60	9.5

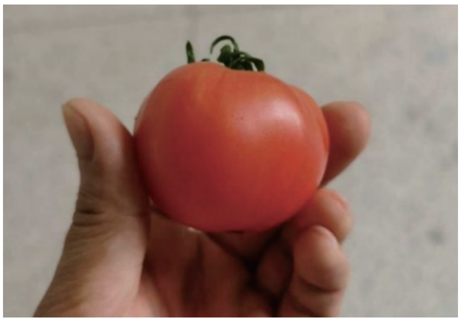


Figure 14 Tomato sample with intact peduncle

Table 7 presents a performance comparison of different picking methods, evaluated primarily based on two indicators: picking success rate and picking time. Overall, the rotational picking method[14] and nested method[16] exhibited lower success rates and longer picking times, showing certain limitations. The airbag clamping method[15] and suction clamping method[17] improved success rates but still had relatively long picking times. In contrast, the method proposed in this study demonstrates superior performance in both success rate and efficiency, with a success rate of 83.3% and an average picking time of 9.5 seconds, showing higher efficiency and stability, making it suitable for greenhouse tomato picking tasks.

Table 7 Performance comparison of picking methods

Picking method	Picking success rate/%	Picking time/s
Rotational picking	60.0	23.0
Airbag clamping	83.9	24.0
Nested picking	57.5	14.9
Suction clamping	72.1	14.6
The method proposed in this study	83.3	9.5

4 Conclusion

This study designs a six-degree-of-freedom tomato-picking robot for greenhouse cultivation environments, achieving autonomous picking operations. To prevent the tomatoes from being damaged by compression during the picking process and to retain the fruit peduncle, a non-contact cavity-type end effector is designed, and the robotic arm's motion is simulated. Additionally, an improved YOLO v8n tomato detection model is proposed, and a comparative analysis is conducted between the improved and the original YOLO v8n models under five different lighting conditions. Finally, field experiments are carried out to validate the effectiveness of the robot prototype in tomato picking. The main conclusions of this study are as follows:

1) The improved YOLO v8n model demonstrated excellent performance, achieving precision, recall, and mAP@0.5 metrics of 94.1%, 89%, and 94.8%, respectively. By introducing a lightweight convolution module, the model size was reduced by 34.3%, enhancing computational efficiency. Furthermore, under five different lighting conditions, the improved YOLO v8n model outperformed the original YOLO v8n model in both average recognition accuracy and detection speed, with the best recognition performance under lighting conditions of 20 000-30 000 lx, exhibiting strong robustness.

2) Tomato localization tests indicate that different lighting intensities can interfere with depth distance measurements. The best measurement accuracy occurs under lighting conditions of 20 000-30 000 lx, where the light is sufficient but not overly strong, optimizing the performance of the depth camera. The test results show that the absolute error between the stereo depth camera and the manually measured target tomato is 1.55 cm, with a relative error of 3.1%.

3) Performance testing for tomato picking was conducted in the field. The robot first identifies and locates the target tomato using the stereo depth camera, then moves the robotic arm to the picking point, and finally controls the end effector to complete the picking task. Experimental results show that the tomato-picking robot achieved a success rate of 83.3%, with an average picking time of 9.5 seconds per tomato, outperforming the 80% success rate of most commercially available picking robots. To further enhance the performance of the robot, future work will consider the spatial distribution of tomato peduncles and their spatial posture to assist the end effector in intelligent obstacle avoidance, thereby improving overall picking efficiency. Additionally, by incorporating real-time feedback from the visual system, the robot can intelligently adjust its path based on environmental changes, minimizing collisions and improving picking success rates and accuracy. These technologies will enhance the robot's adaptability and performance in complex environments.

Acknowledgements

The work was supported by State Key Laboratory of Precision Blasting and Hubei Key Laboratory of Blasting Engineering, Jianghan University (Grant No. BL2021-16), and the Natural Science Foundation of Hubei Province (Grant No. 2025AFB176).

Code and data availability

The program code is available at the following link: https://github.com/henizaba/SC-YOLO. Due to data privacy concerns, the full dataset cannot be publicly disclosed; however, limited samples can be provided upon reasonable request.

[References]

[1] Li J M, Xiang C Y, Wang X X, Guo Y M, Huang Z J. Current status and prospects of China's tomato industry during the 13th Five-Year Plan. China Vegetables, 2021;(2): 13–20. (in Chinese)

[2] Zhou M, Li C B. The current status and prospects of tomato seed industry in China. Vegetables, 2022;(5): 6–10. (in Chinese)

[3] Hu H R, Zhang Y Y, Zhou J L, Chen Q, Wang J P. Research status of end effectors for fruit and vegetable harvesting robots. Chinese Journal of Agricultural Machinery, 2024; 45(4): 231–236. (in Chinese)

[4] Zhang P, Tang Q Y, Chen L F, Qin C L, Wang L Y. Research status and progress analysis of fruit and vegetable picking robots. Southern Agricultural Machinery, 2023; 54(7): 9–12.

[5] Molaei F, Ghatrehsamani S. Kinematic-based multi-objective design optimization of a grapevine pruning robotic manipulator. AgriEngineering, 2022; 4(3): 606–625.

[6] Moreira G, Magalhães S A, Pinho T, dos Santos F S, Cunha M. Benchmark of deep learning and a proposed HSV colour space models for the detection and classification of greenhouse tomato. Agronomy, 2022; 12(2): 356.

[7] Huang C W, Cai D X, Wang W Z, Li J, Duan J L, Yang Z. Development of an automatic control system for a hydraulic pruning robot. Computers and Electronics in Agriculture, 2023; 214: 108329.

[8] Zhao Q, Li L J, Wu Z C, Guo X, Li J. Optimal design and experiment of manipulator for camellia pollen picking. Applied Sciences, 2022; 12(16): 8011.

[9] Chen R. Application and development trends of mechanical automation technology in modern agriculture. Agricultural Engineering Technology, 2024; 44(2): 64–65. (in Chinese)

[10] Li T H, Sun M, He Q H, Zhang G S, Shi G Y, Ding X M, et al. Tomato recognition and location algorithm based on improved YOLOv5. Computers and Electronics in Agriculture, 2023; 208: 107759.

[11] Kondo N, Yata K, Iida M, Shiigi T, Monta M, Kurita M, et al. Development of an end-effector for a tomato cluster harvesting robot. Engineering in Agriculture, Environment and Food, 2010; 3(1): 20–24.

[12] Liu J Z, Peng Y, Faheem M. Experimental and theoretical analysis of fruit plucking patterns for robotic tomato harvesting. Computers and Electronics in Agriculture, 2020; 173: 105330.

[13] Liu J Z, Yuan Y, Gao Y, Tang S Q, Li Z G. Virtual model of grip-and-cut picking for simulation of vibration and falling of grape clusters. Transactions of the ASABE, 2019; 62(3): 603–614.

[14] Yaguchi H, Nagahama K, Hasegawa T, Inaba M. Development of an autonomous tomato harvesting robot with rotational plucking gripper. In: 2016 IEEE/RSJ international conference on intelligent robots and systems (IROS), Daejeon, Korea (South): IEEE, 2016; pp.652–657.

[15] Wang X N, Wu P H, Feng Q C, Wang G H. Design and experiments of tomato picking robot system. Agricultural Mechanization Research, 2016; 38(4): 94–98. (in Chinese)

[16] Zheng X J, Rong J C, Zhang Z Q, Yang Y, Li W, Yuan T. Fruit growing direction recognition and nesting grasping strategies for tomato harvesting robots. Journal of Field Robotics, 2023; 41(2): 300–313.

[17] Rong J C, Wang P B, Wang T J, Hu L, Yuan T. Fruit pose recognition and directional orderly grasping strategies for tomato harvesting robots. Computers and Electronics in Agriculture, 2022; 202: 107430.

[18] Fujinaga T, Yasukawa S, Ishii K. Development and evaluation of a tomato fruit suction cutting device. In: 2021 IEEE/SICE International Symposium on System Integration (SII), Iwaki, Fukushima, Japan: IEEE, 2021; pp.628–633. doi: 10.1109/IEEECONF49454.2021.9382670.

[19] Liang X F, Jin C Q, Ni M D, Wang Y W. Acquisition and experiment on location information of picking point of tomato fruit clusters. Transactions of the Chinese Society of Agricultural Engineering, 2018; 34(16): 163–169. (in Chinese)

[20] Feng J H, Li Z W, Rong Y L, Sun Z L. Identification of mature tomatoes based on an algorithm of modified circular Hough transform. Chinese Journal of Agricultural Machinery and Chemistry, 2021; 42(4): 190–196. (in Chinese)

[21] Li H, Tao H X, Cui L H, Liu D W, Sun J T, Zhang M. Tomato fruit recognition and localization method based on SOM-K-means algorithm. Transactions of the Chinese Society for Agricultural Machinery, 2021; 52(1): 23–29. (in Chinese)

[22] Wang J P, Xu G. Research on tomato maturity detection method based on machine vision and electronic nose fusion. Food and Machinery, 2022; 38(02): 148–152.

[23] Benavides M, CantónGarbín M, SánchezMolina J A, Rodríguez F. Automatic tomato and peduncle location system based on computer vision for use in robotized harvesting. Applied Sciences, 2020; 10(17): 5887.

[24] Fass E, Shlomi E, Ziv C, Glikman O, Helman D. Machine learning models based on hyperspectral imaging for pre-harvest tomato fruit quality monitoring. Computers and Electronics in Agriculture, 2025; 229: 109788. doi: 10.1016/J.COMPAG.2024. 109788.

[25] Bai Y H, Mao S H, Zhou J, Zhang B H. Clustered tomato detection and picking point location using machine learning-aided image analysis for automatic robotic harvesting. Precision Agriculture, 2023; 24(2): 727–743.

[26] Rong J C, Zhou H, Zhang F, Yuan T, Wang P B. Tomato cluster detection and counting using improved YOLOv5 based on RGB-D fusion. Computers and Electronics in Agriculture, 2023; 207: 107741.

[27] Zhang R C, Zhou Y C, Hou Y H, Liu Z Y, Zhao H G, Zhao Y H. Counting tomatoes with different maturities using ultra-depth masking and improved

YOLOv8. Transactions of the Chinese Society of Agricultural Engineering, 2024; 40(24): 146–156. (in Chinese)

[28] Wang X W, Liu J. Tomato anomalies detection in greenhouse scenarios based on YOLO-Dense. Frontiers in Plant Science, 2021; 12: 634103.

[29] Li T H, Sun M, Ding X M, Li Y H, Zhang G S, Shi G Y, et al. T Tomato recognition method at the ripening stage based on YOLO v4 and HSV. Transactions of the Chinese Society of Agricultural Engineering, 2021; 37(21): 183–190. (in Chinese)

[30] Wang X F, Wu Z W, Jia M, Xu T, Pan C L, Qi X B, et al. Lightweight SM-YOLOv5 tomato fruit detection algorithm for plant factory. Sensors, 2023; 23(6): 3336.

[31] Cai Y Q, Cui B, Deng H, Zeng Z, Wang Q C, Lu D J, et al. Cherry tomato detection for harvesting using multimodal perception and an improved YOLOv7-tiny neural network. Agronomy, 2024; 14(10): 2320.

[32] Miao R H, Li Z W, Wu J L. Lightweight cherry tomato maturity detection method based on improved YOLO v7. Transactions of the Chinese Society of Agricultural Machinery, 2023; 54(10): 225–233. (in Chinese)

[33] Huang Y P, Liu Y, Yang Y T, Zhang Z W, Chen K J. Assessment of tomato color by spatially resolved and conventional vis NIR spectroscopies. Spectroscopy and Spectral Analysis, 2019; 39(11): 3585–3591. (in Chinese).

[34] Zhang Z L L, He T T, Li Z W, Shi K L, Liu C B, Zheng W G. Quantitative grading method for tomato maturity using regional brightness correction. Transactions of the Chinese Society of Agricultural Engineering, 2023; 39(7): 195–204. doi: 10.11975/j.issn. 1002-6819.202211192. (in Chinese)