FASE FRONTIERS OF AGRICULTURAL SCIENCE AND ENGINEERING
Engineering Transactions of CAE
RESEARCH ARTICLE
2026 / Volume 13 / Issue 3 / 25657

Tomato ripeness detection method based on improved YOLOv11 lightweight model

Dongyang WANG[1], Zhijie FANG (✉)[1], Man MO[2], Jinchong GAN[1], Zijun SUN (✉)[1]

1 School of Electronics Engineering, Guangxi University of Science and Technology, Liuzhou 545006, China.
2 School of Science, Guangxi University of Science and Technology, Liuzhou 545006, China.

KEYWORDS

C2PSA_SCSA, C3k2_MSCB, HS-FPN, lightweight, tomato ripeness detection, YOLOv11n

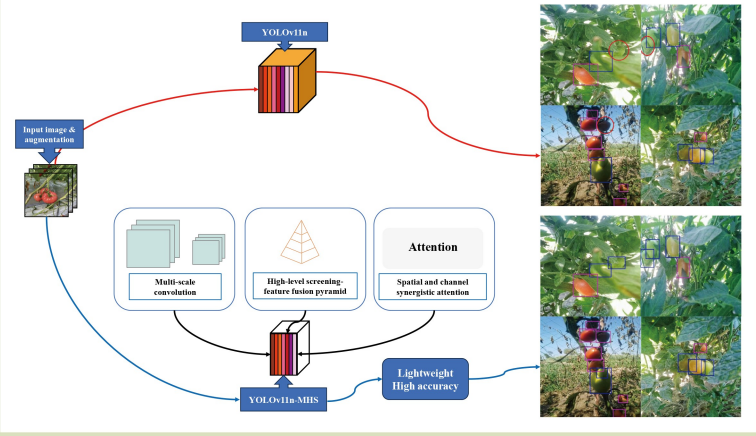
HIGHLIGHTS

- A lightweight YOLOv11-MHS model is proposed for tomato ripeness detection.
- A key feature is the inclusion of a module for multiscale feature extraction and fusion.
- The model is structured for fusion, boosting robustness and making it lightweight.
- Another key module integrates spatial and channel synergistic attention for complex scene adaptation.
- The new model reduces parameters and size by about a third compared to YOLOv11n.

Received June 5, 2025;
Accepted August 22, 2025.

Correspondences: nnfang@semi.ac.cn, sunzijun@gxust.edu.cn

GRAPHICAL ABSTRACT



ABSTRACT

To address the challenges faced in real-world tomato ripeness detection, such as variable lighting conditions, complex backgrounds, and the trade-off between accuracy and the model being effectively lightweight, this study proposes a lightweight YOLOv11-MHS model. The improvements of the proposed model are reflected in three aspects: (1) the C3k2_MSCB module is designed, which integrates a multiscale convolutional block (MSCB) for multiscale feature extraction and fusion, thereby enhancing detection accuracy; (2) the neck of the model is redesigned as a high-level feature screening-fusion pyramid structure, which fuses key features to improve robustness in cluttered environments while reducing model size; and (3) the C2PSA module is enhanced by introducing the spatial and channel synergistic attention mechanism to improve the ability of the model to handle complex scenes. Experimental results on the same data set show that, compared to the baseline model YOLOv11n, YOLOv11-MHS achieves improvements of 1.7% in mAP0.5 and 2.9% in mAP0.5-0.95, while reducing parameters and model size by 35.2% and 32.7%, respectively. These results demonstrate that YOLOv11-MHS achieves both outstanding accuracy and lightweight performance in tomato ripeness detection, providing technical support for agricultural applications.

1 Introduction

Tomato is one of the most widely cultivated and economically important horticultural crops globally, owing to its high nutritional value and rich content of bioactive compounds, such as lycopene and vitamin C[1]. It has high yield potential under diverse agroecological conditions and can be cultivated in both open-field systems and controlled environments, such as greenhouses[2,3]. Unlike fruits that ripen uniformly, tomatoes exhibit asynchronous maturation, leading to the concurrent presence of fruits at various ripeness stages on a single plant. Given the relatively short postharvest shelf life of tomatoes[4], precise and timely assessment of ripeness is critical for minimizing postharvest losses and maintaining product quality throughout the supply chain.

According to statistics, the average postharvest loss of fruit in China has reached 20%, inflicting an annual economic loss exceeding 100 billion CNY. In contrast, developed countries typically record losses below 5%, with some achieving as little as 1% to 2%. A primary cause of this disparity is the asynchronous maturation of fruit, which frequently leads growers to misjudge optimal harvest windows and, consequently, to premature or delayed picking[5]. In greenhouse tomato production, labor costs alone account for more than 44.5% of the net profit[6].

In natural environments, tomato maturity detection presents several challenges: fruits at varying ripeness stages coexist on the same plant and fluctuating illumination significantly impairs recognition accuracy. Also, dense foliage occlusion, along with the complex background of soil and weeds, frequently obscures fruit features, leading to erroneous maturity assessments. At present, the harvesting and sorting of tomatoes predominantly depend on manual labor[7]. As tomato production continues to rise, manual harvesting methods are increasingly plagued by issues such as low efficiency, high labor intensity and escalating costs[8].

Recent advances in artificial intelligence and deep learning have facilitated the widespread application of object-detection techniques in agricultural production, including crop-disease diagnosis[9–12] and fruit detection[13]. These techniques have been applied to various fruit types, including apples, citrus, strawberries, and kiwifruit, facilitating automated detection and localization of fruit in agricultural settings[14–16]. By replacing visual observation with automated vision-based systems, these methods contribute to improving operational efficiency and standardizing fruit quality assessment in modern agricultural systems.

In recent years, the application of deep learning in fruit ripeness detection has become a research hotspot. Zheng et al.[17] proposed a YOLOv8-Tomato model based on the improved YOLOv8 architecture. By introducing the LSKA attention mechanism, adopting the Dysample dynamic upsampler, and replacing the inner-IoU loss function, the model achieved a Mean Average Precision at 0.5 IoU threshold of 99.4% in recognizing tomato ripeness in greenhouse environments. Hu et al.[18] improved YOLOv5s by introducing the Ghostconv backbone network, CA attention mechanism, and BiFPN feature fusion structure, resulting in an accuracy of 90.1% for recognizing ripe tomatoes. Wang et al.[19] developed the YOLOv9-C model, which is targeted at tomato detection in greenhouse environments, achieving an accuracy of 97.2% and a recall rate of 92.3%. Ma and Zhou[20] proposed an improved YOLOv7 model, which realizes the detection of grape ripeness. The precision of this model on the grape image test set reaches 95.2%, with a memory footprint of 53.6 MB. Xie et al.[21] proposed a strawberry ripeness detection model named YOLO-SR, which is improved based on YOLOv5s. This model achieves accurate recognition of strawberry fruits in greenhouse environments, with a mean average precision of 98.0% on the self-built data set. Majdudin[22] developed a computer vision system using the YOLO v8 algorithm for pineapple ripeness detection, and the model achieved an accuracy of 84.8%. Parvathi et al.[23] integrated the ResNet-50 backbone network into Faster-RCNN for coconut ripeness detection in complex backgrounds, and its performance is superior to that of mainstream single-stage detectors such as SSD, YOLOv3 and R-FCN.

Although existing models have delivered notable success in fruit detection, several limitations remain. Some models detect only the presence of fruit without accurately assessing ripeness levels, which limits their applicability for maturity-specific tasks. Many approaches have been primarily validated in controlled greenhouse environments or on harvested fruits, leading to suboptimal performance when applied to complex outdoor settings characterized by variable lighting and background conditions. Also, a considerable number of these models require substantial memory and computational resources.

To mitigate the adverse effects of variable illumination and occlusion on tomato maturity detection accuracy, this study proposes a lightweight maturity-assessment model, termed YOLOv11-MHS. The architecture is designed to achieve high detection precision while maintaining relatively small memory consumption, thereby enhancing robustness under complex agricultural conditions and enabling more efficient and reliable ripeness classification.

2 Materials and methods

2.1 YOLOv11 model improvement

Tomato ripeness detection is a complex visual task that fundamentally depends on discriminative features such as color, shape and texture, each exhibiting varying levels of significance across different spatial scales. Additionally, external factors such as varying lighting conditions, complex backgrounds, and partial occlusions often introduce substantial challenges to the overall performance of detection models. To tackle these challenges, this paper proposes a novel improved YOLOv11-MHS model for tomato ripeness detection. A multiscale convolution block (MSCB)[24] is integrated into the C3k2 module of the backbone, allowing simultaneous capture of fine and coarse features through multiscale and depthwise separable convolutions, thereby enabling more comprehensive extraction of ripeness-relevant visual cues. This enhances feature representation while maintaining computational efficiency. In addition, the neck of the network is restructured using a high-level screening-feature fusion pyramid (HS-FPN)[25]. This design enhances feature fusion by hierarchically aggregating information from multiple scales, while simultaneously screening out redundant or noisy features. Such fusion improves robustness against environmental variability and contributes to a more compact model with reduced parameter count and computational overhead. Also, to further improve the adaptability of the model to real-world scenarios, a spatial and channel synergistic attention (SCSA)[26] is introduced into the C2PSA module. By jointly modeling spatial and channel-wise dependencies, SCSA facilitates the selective enhancement of salient regions and feature channels. This synergistic attention mechanism significantly boosts the ability of the model to distinguish ripe tomatoes from visually similar background elements, particularly under challenging conditions such as uneven illumination, occlusion or background clutter. The network architecture of the improved YOLOv11-MHS model is shown in Fig. 1.

2.2 C3k2_MSCB

Tomato ripeness detection demands real-time accuracy, which requires monitoring color, shape, and texture changes. However, the detection process is significantly affected by visually challenging conditions like uneven lighting, occlusion and background noise. To address these challenges, this study uses the C3k2_MSCB module to improve detection accuracy.

The MSCB uses multiscale depthwise convolution (MSDC) to extract features across different scales, capturing tomatoes of various sizes and shapes. As shown in Fig. 2, MSCB improves detection accuracy by fusing multiscale features, even in cases of partial occlusion.

Lighting changes can impact color information. The MSCB captures color information more effectively by integrating interchannel relationships. This enhances the robustness to lighting changes, ensuring high detection accuracy in tough lighting conditions. In challenging agricultural settings, tomato backgrounds are often disturbed by factors like branches and soil, which can affect feature extraction. The MSCB refines feature maps through multiscale convolutions. It emphasizes the shape and texture of tomatoes while suppressing background noise, thus improving the accuracy of tomato maturity detection in complex background scenarios. The MSCB integrates interchannel relationships through a cascaded architecture composed of pointwise convolutions, MSDC, and channel shuffling.

Illumination changes primarily disturb the absolute intensities of color channels; for example, increased brightness raises all RGB channel values globally. Nevertheless, relative interchannel statistics such as ratios and differences remain notably stable. Regardless of illumination intensity, the red channel value of a ripe tomato consistently exceeds its green channel value.

As illustrated in Fig. 2(b), the first module of MSCB is a 1×1 pointwise convolution that expands the channel count. Figure 2(a) reveals that MSDC comprises several parallel depthwise convolutions, each using a distinct kernel size and followed immediately by batch normalization and ReLU6. By applying depthwise convolutions in parallel across multiple scales, MSDC captures multiscale features. The batch normalization layers mitigate illumination-induced intensity fluctuations by normalizing each channel.

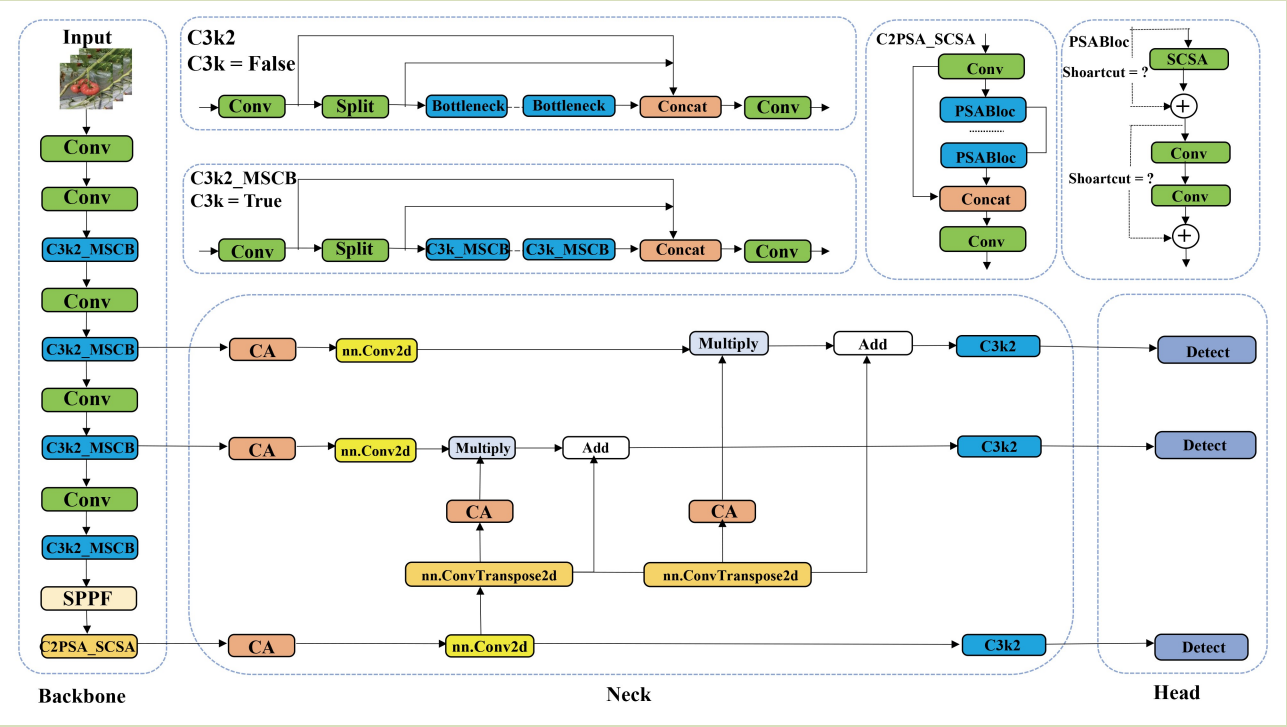


Fig. 1 YOLOv11-MHS network architecture. C3k = True indicates the integration of the multiscale convolution block (MSCB); and C3k = False represents the standard C3 block without MSCB integration.

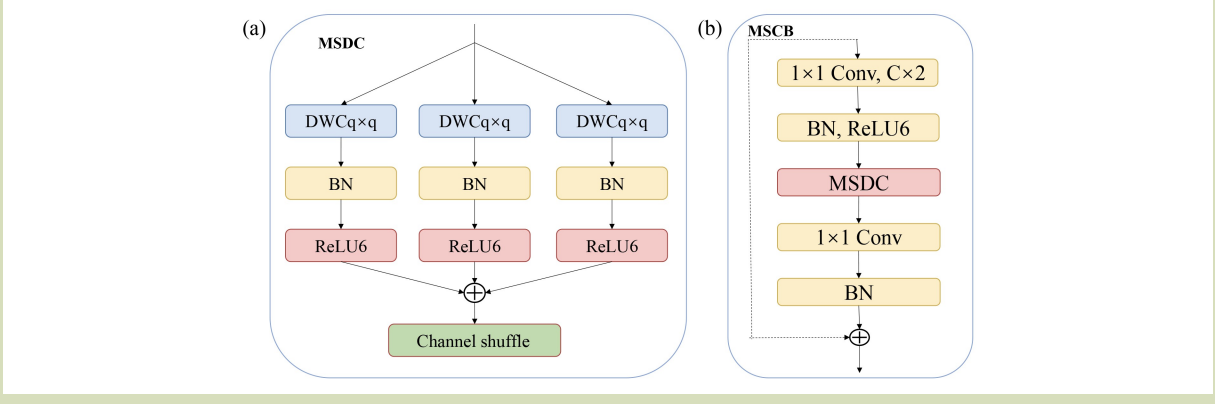


Fig. 2 Structure of (a) multiscale convolution block (MSDC) and (b) multiscale depthwise convolution (MSCB).

Subsequent to MSDC, a channel shuffle operation, annotated in Fig. 2(a), rearranges and redistributes the grouped channels. The shuffled output, together with the MSDC result, is then compressed back to the original channel count via a 1 × 1 pointwise convolution.

MSDC decomposes a standard convolution into depthwise and pointwise components, yielding a significant parameter reduction, with C_{in} and C_{out} denoting the input and output channel counts, respectively, and $K \times K$ the kernel size. The parameter count of a standard convolution is $C_{in} \times C_{out} \times K \times K$

whereas that of depthwise separable convolution is $C_{in}\times 1\times K\times K+C_{in}\times C_{out}\times 1\times 1$. This reduction directly lowers the memory footprint occupied by the weights of the model.

2.3 High-level screening-feature fusion pyramid

HS-FPN is incorporated into the neck layer. By fusing multiscale features and combining high-level and low-level features, HS-FPN ensures the accurate extraction of features for tomatoes of different sizes. The channel attention (CA) module in HS-FPN filters out useful features for ripeness detection while suppressing irrelevant ones. This fusion and strengthening of key features improves the anti-interference capability of the model in cluttered scenarios. Additionally, HS-FPN reduces model parameters and computational cost.

HS-FPN comprises two key components: a feature selection module and a feature fusion module.

The feature selection module uses CA and dimension matching mechanisms to screen feature maps at different scales. The CA module first performs global average pooling and global max pooling on the input feature maps. Subsequently, the two descriptive vectors are merged and mapped via the Sigmoid function to obtain $f_{CA}\in R^{C\times 1\times 1}$, which represents the attention weight for each channel.

Multiplying these weights with the corresponding scale feature maps generates the filtered feature maps. The dimension matching module uses 1 × 1 convolution to reduce the number of channels in each scale feature map to 256. This value is fixed and such a design aims to keep the number of channels consistent across feature maps of different levels, thereby facilitating subsequent feature fusion operations. HS-FPN needs to perform cross-level feature addition, and unifying the number of channels to 256 ensures that features from all levels can be directly added. Using 256 channels can achieve a good balance between accuracy and efficiency. Using too few channels will limit the expressive capability of features, while using an excessive number of channels will lead to a significant increase in model parameters and computational load.

Feature fusion module: in the multiscale feature maps generated by the backbone network, high-level features are rich in semantic information but less precise in object localization whereas low-level features are precise in localization but limited in semantic information. Therefore, a selective feature fusion mechanism is adopted to combine the filtered low- and high-level features. After high-level features are expanded and upsampled via bilinear interpolation or transposed convolution, they are fused with low-level features to boost the ability of the model to represent tomato ripeness features.

Given an input high-level feature and a low-level feature, the input high-level feature is denoted as $F_{high}\in R^{C\times H\times W}$, where, C represents the number of channels and $H\times W$ is the spatial resolution. The input low-level feature is denoted as $F_{low}\in R^{C\times H1\times W1}$, where, $H1\times W1$ stands for the spatial resolution. The high-level feature is first expanded using a transposed convolution with a stride of 2 and a kernel size of 3×3, resulting in a feature size of $F_{\widehat{high}}\in R^{C\times 2H\times 2W}$. The spatial resolution is extended from $H\times W$ to $2H\times 2W$, laying the groundwork for subsequent matching with low-level features. To align the dimensions of the high-level and low-scale features, high-level features are upsampled or downsampled via bilinear interpretation get the feature $F_{att}\in R^{C\times H1\times W1}$. F_{att} is the high-level feature after size alignment, where, $H1$ and $W1$ represent the height and width of the low-level feature F_{low} (F_{att} is completely matched with F_{low} in terms of spatial dimensions).

Subsequently, F_{att} is fed into the CA channel attention module to generate channel attention weights, which in turn perform pointwise weighted filtering on the low-level feature F_{low}. Then finally, the filtered low-scale features are fused with the high-level ones to boost the feature representation of the model, get $F_{out}\in R^{C\times H1\times W1}$. These equations illustrate the fusion process:

$$F_{att}=BL(T-Conv(F_{high})) \quad (1)$$

$$F_{out}=F_{low}\times CA(F_{att})+F_{att} \quad (2)$$

where, $CA(\cdot)$ is the weights generated by channel attention, and × represents the pointwise multiplication operation. First, the high-level features are used to perform attention-based filtering on the low-level features; then, the two are added under the same resolution, so as to improve the detection or classification performance. The structure of HS-FPN is shown in Fig. 3.

2.4 C2PSA_SCSA

Tomato ripeness detection involves not only identifying color variations but also capturing morphological features such as shape, size and contextual background cues. These visual

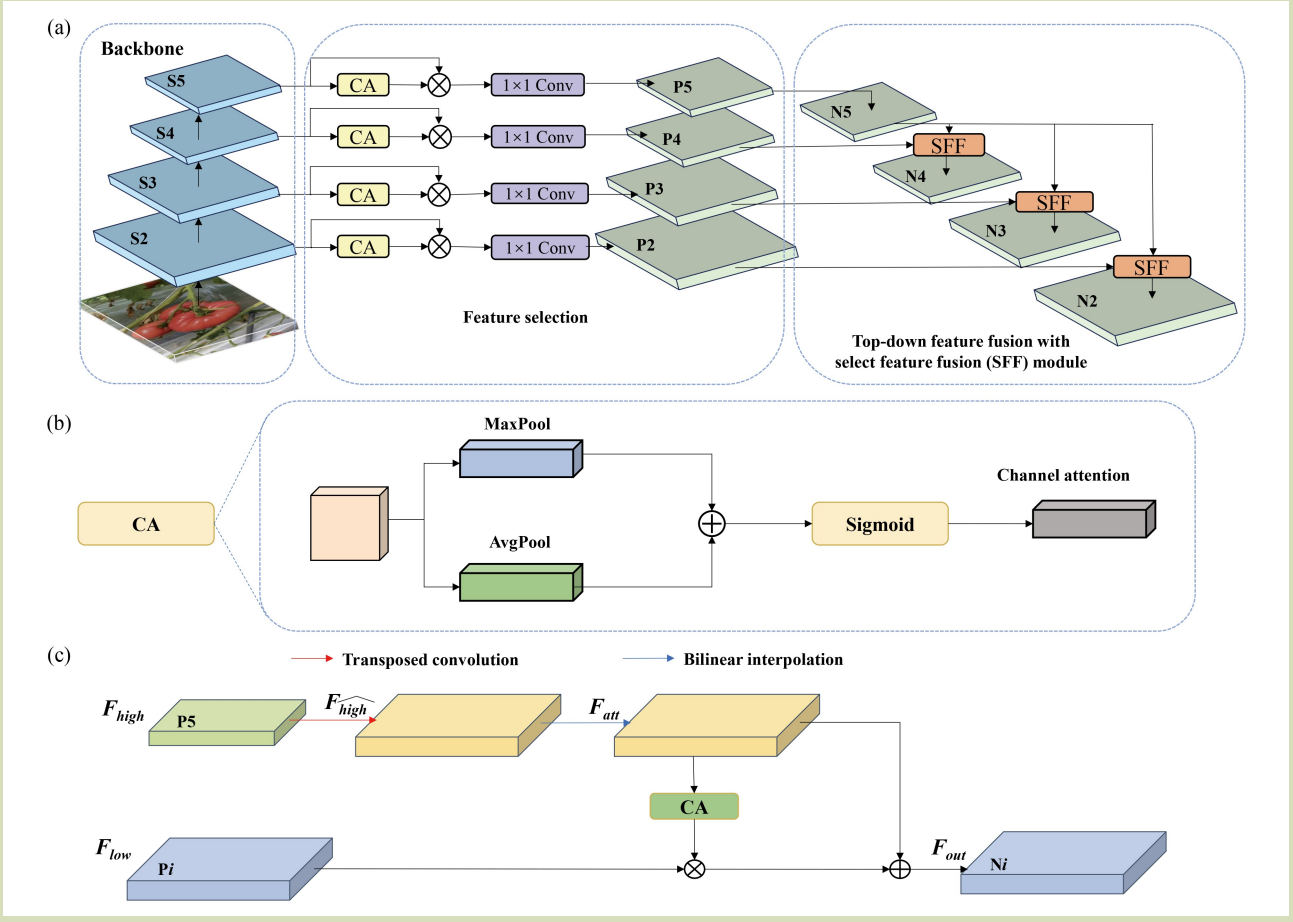


Fig. 3 Structure of high-level screening-feature fusion pyramid network (HS-FPN). (a) HS-FPN; (b) structure diagram of channel attention module (CA); (c) structure diagram of selective feature fusion module (SFF).

attributes contain diverse semantic information distributed across both spatial and channel dimensions. In particular, color channels have differentiated purposes in ripeness classification: the red channel is essential for highlighting ripe tomatoes, while the green channel is more informative for identifying unripe fruits. Also, spatial attention mechanisms are critical for enhancing feature representation related to object shape and positional context, which is especially beneficial in scenarios involving occlusion or cluttered backgrounds.

To address these challenges, this study designs a novel C2PSA_SCSA module by integrating the SCSA mechanism into the original C2PSA module. The SCSA mechanism is composed of two key components: shared multisemantic spatial attention (SMSA) and progressive channel self-attention (PCSA). The SMSA module aggregates multilevel semantic spatial information and uses it to guide channel-wise attention, thereby enabling more focused and effective extraction of key spatial features.

The SMSA decomposes the given input $X \in R^{(B\times C\times H\times W)}$ (where, B is the batch size, C is the number of channels, and H and W are the height and width of the feature map, respectively) along the height and width dimensions. By applying global average pooling along each dimension, it generates two unidirectional one-dimensional sequence structures: $X_H \in R^{(B\times C\times W)}$ and $X_W \in R^{(B\times C\times H)}$. To learn diverse spatial distributions and contextual relationships, the feature set is partitioned into four independent sub-features of equal size, each with $C/4$ channels. To extract multiscale spatial information within each sub-feature, depthwise 1D convolutions with kernel sizes of 3, 5, 7 and 9 are applied respectively.

Corresponding sub-features in the two sequence features are aligned via shared convolution kernels. Then finally, group normalization is applied to normalize the sub-features at different scales, followed by the Sigmoid activation function to generate the multisemantic spatial attention information. In parallel, the PCSA module progressively enhances channel attention, helping to strengthen the representation of small or distant tomatoes that are prone to feature loss.

In PCSA, the term "progressive" in "progressively enhancing channel attention" refers to the gradual compression along the spatial dimension. It uses multi-stage average pooling to progressively compress the spatial resolution of the feature maps, rather than performing single-step global pooling.

In PCSA, the feature map output by the SMSA module undergoes average pooling and normalization to obtain a preliminarily compressed feature map. A 1 × 1 two-dimensional depthwise convolution is applied to perform a linear transformation on the input feature map, generating query, key and value. Subsequently, the feature map is processed through an average pooling operation and an activation function (sigmoid), and then fused with the input feature map to yield the output feature map.

By synergistically combining spatial and channel information, the SCSA mechanism enhances the capacity of the network to localize and distinguish tomato targets of varying ripeness with higher precision. It improves feature discrimination, supports small object detection, and strengthens the resilience of the model to environmental variability. The structure of the proposed SCSA module is shown in Fig. 4.

3 Experiment and analysis

3.1 Data sets

3.1.1 Tomato data set

The data set used in this study comprises 1975 tomato images, including 1875 captured at plantations in Liuzhou, Guangxi, China using an IQOO9 Pro (3060 × 3060 resolution, JPEG format) under varied angles and lighting. To augment generalization, 100 images were added from public data sets[27].

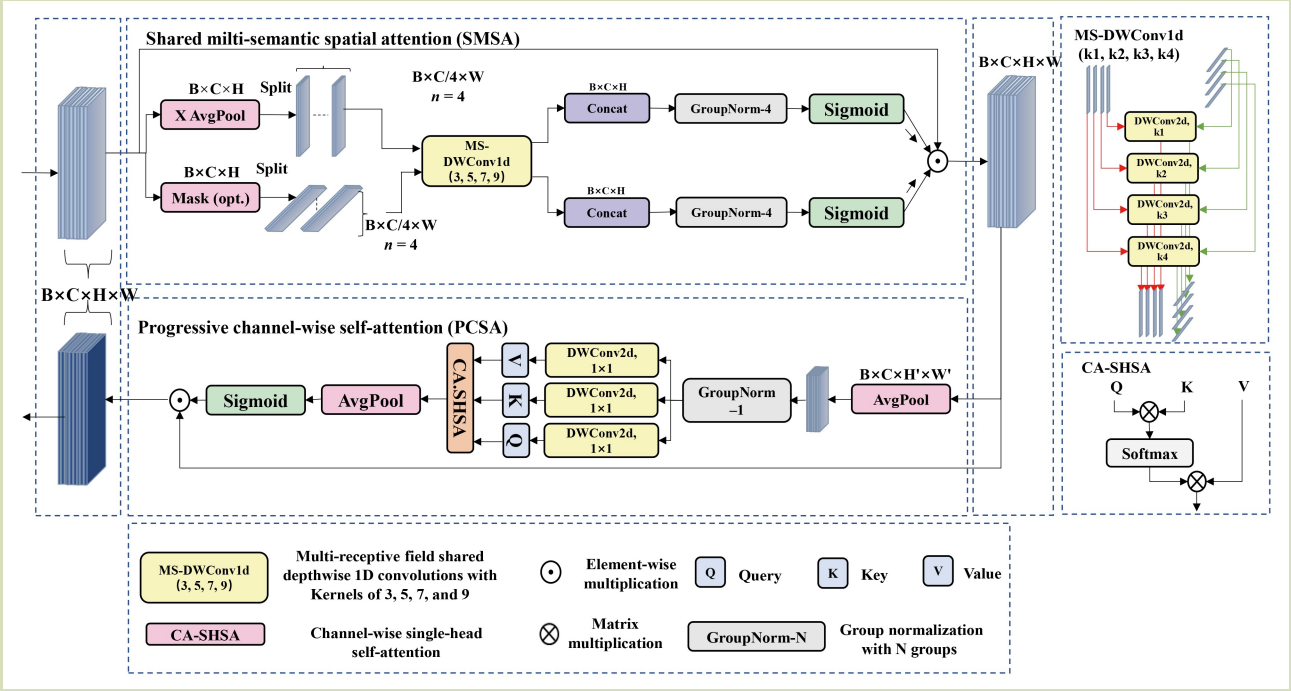


Fig. 4 Structure of spatial and channel synergistic attention (SCSA).

The tomato ripeness detection data set constructed for this study comprises 23,670 valid samples. Phenotypic trait-based classification revealed: ripe fruits accounted for 65.5% (15,494 samples), while unripe fruits constituted 34.5% (8176 samples). Examples of tomatoes in natural scenes and with different ripeness levels are shown in Fig. 5.

The ripening process of tomato fruits can be divided into the green maturity stage, color-breaking stage, mature stage, and full maturity stage, each with distinct fruit characteristics and appropriate harvesting times. At the color-breaking stage, the fruits appear yellowish-green and start to turn red but only slightly, having initially developed their flavor with firm flesh, which makes them suitable for transportation and renders this stage ideal for ripening induction and harvesting. During the maturation stage, over 80% of the fruit or the entire fruit turns red, with the flesh remaining relatively firm and the taste reaching its optimum, making it the ideal period for fresh fruit supply and harvesting. In the full maturity stage, the fruits are fully ripe, with a completely red color and soft flesh, suitable for processing into jam or reserved for seeds[28].

According to research, for fresh consumption, tomatoes should be harvested at the pink or semi-ripe stage. For seed production, fully red-ripe tomatoes are the ideal choice[29]. In consideration of these reasons and the fact that tomatoes have a short postharvest freshness period, ethylene gas from surrounding tomatoes can accelerate the ripening of semi-ripe ones during storage and transport[30], semi-ripe tomatoes are classified as ripe in this data set. Thus, the data set includes a larger number of ripe tomatoes. This merged classification can meet the requirements of most application scenarios in daily life; however, in scenarios requiring highly precise ripeness classification, manual intervention may be necessary.

3.1.2 Data creation and augmentation

In the data creation and augmentation phase, images are first resized to a uniform dimension of 640 × 640 pixels, and then the LabelImg tool is used for sample annotation. To enhance the adaptability of the model to diverse scenarios, data augmentation techniques are used to expand the initial training set from 1383 images to 3911. These augmentation methods include flipping, rotation, brightness and contrast adjustment, cropping and noise addition. Specifically, flipping involves horizontal mirroring across the vertical axis and combined flipping in both horizontal and vertical directions. Rotation is performed clockwise in 90° increments. Brightness adjustment is designed to mimic the effects of varying lighting conditions on images, whereas contrast adjustment serves to highlight the differences between tomatoes and the background or other objects. Noise addition is used to simulate various types of disturbances that images might encounter in real-world scenarios. Finally, cropping is applied to increase the randomness of the data set (Fig. 6 shows examples).

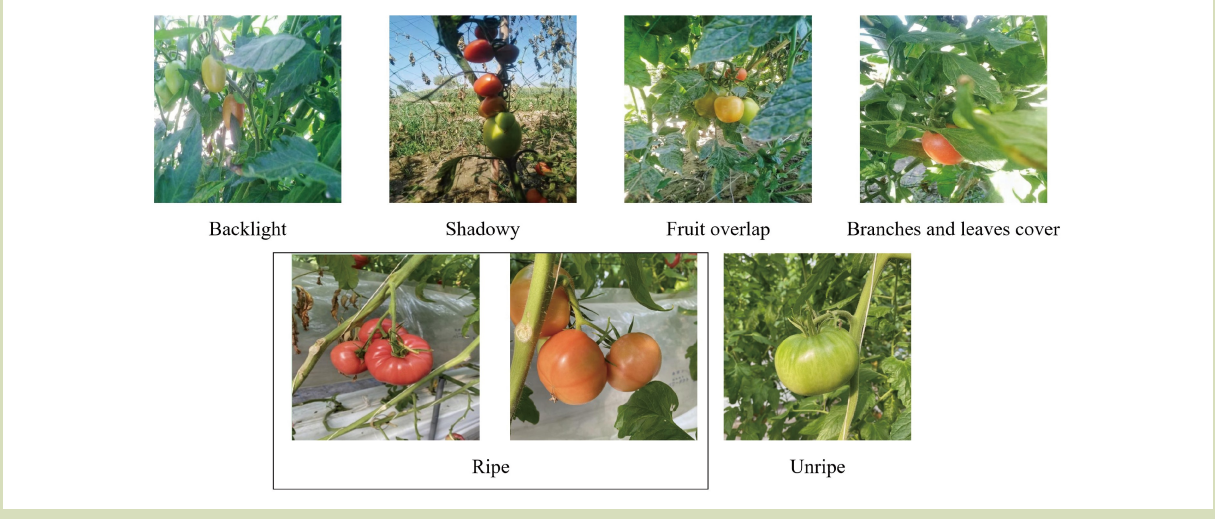


Fig. 5 Examples of tomatoes in natural scenes and with different ripeness levels.

Fig. 6 Tomato image enhancement examples.

3.2 Experimental environment configuration and evaluation metrics

Experiments in this study were conducted in a server environment featuring an Intel Xeon Silver 4214R processor (12 cores @ 2.40 GHz), an NVIDIA GeForce RTX 3080 Ti graphics accelerator (12 GB GDDR6X VRAM) and 90 GB memory. The software stack used included the Ubuntu 20.04 operating system with Python 3.9.21, the PyTorch 1.31.1 framework and CUDA 11.6 acceleration libraries.

The training protocol used a stochastic gradient descent optimizer, with an initial learning rate of 0.01, a momentum coefficient of 0.937 and a weight decay of 0.0005 with the batch size set to 16. As can be seen in Fig. 7, when the number of model iterations (epochs) reaches 250, the mAP value becomes stable. Therefore, the number of model iterations is set to 250.

For tomato ripeness detection in real-world scenarios, achieving high model accuracy is essential. To comprehensively evaluate the performance of the model, this study uses several key metrics: precision, recall, mean average precision (mAP), parameters, GFLOPs and memory size. Model is assessed as lightweight through parameters, GFLOPs and memory size. For accuracy, the main metrics are precision, recall and mAP. The metric mAP is further divided into mAP0.5 and mAP0.5-0.95 with mAP0.5 representing the mean average precision at an IoU threshold of 0.5 and mAP0.5-0.95 the average mAP calculated as the IoU threshold increases from 0.5 to 0.95 in steps of 0.05. These four metrics were calculated as:

$$\mathrm{P} = \frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FP}} \tag{3}$$

$$R = \frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FN}} \tag{4}$$

$$\mathrm{AP} = \int_0^1 p(r)dr \tag{5}$$

$$\mathrm{mAP} = \frac{1}{n}\sum_{i=1}^{n}\mathrm{AP}_i \tag{6}$$

where, TP is the number of positive samples correctly

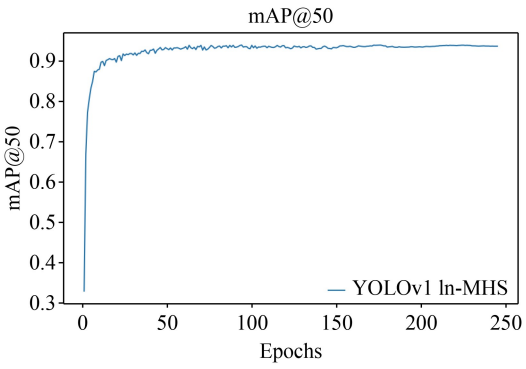


Fig. 7 Process of model training.

identified, TN is the number of negative samples accurately classified, FP is the number of negative samples misclassified as positive and FN, is the number of positive samples misclassified as negative.

3.3 Ablation experiment

To systematically evaluate the performance contributions of each structural enhancement in the proposed YOLOv11-MHS model for tomato ripeness detection, we conducted seven ablation experiments on a unified data set, and progressively compared various improvements against the YOLOv11n baseline. Table 1 summarizes the quantitative results of these experiments, with the first row representing the performance metrics of the unaltered YOLOv11n baseline. The "√" indicates the module used in the respective group.

In experiment 1, the modified C3k2 module enhanced precision, mAP0.5 and mAP0.5-0.95 while reducing the number of parameters, GFLOPs and memory size. This improvement is attributed to the ability of the C3k2_MSCB module to simultaneously capture multiscale features, thereby enhancing tomato detection performance. Also, by integrating interchannel relationships, the module can more effectively handle illumination variations, which further improves the robustness of the model. In experiment 2, redesigning the neck with HS-FPN decreased the number of parameters by 736,912 (to 1,845,630) and memory size by 1.5 MB. This is primarily attributed to the neck structure using HS-FPN and this optimization achieved a substantial reduction in computational cost and model size. In experiment 3, combining SCSA with C2PSA (forming C2PSA_SCSA) improved recall by 2% and also increased mAP0.5 and mAP0.5-0.95. This indicates that C2PSA_SCSA can more effectively extract maturity-related features and suppress background interference. Specifically, the SCSA mechanism, through SMSA and PCSA, enhances the ability of the model to localize and distinguish tomatoes at different ripeness stages. In experiment 4, simultaneous modifications of the C3k2 module and neck structure improved precision, recall and mAP, while reducing the number of parameters and memory size. In experiment 5, modifying the C3k2 module and C2PSA increased precision by 1.3%, recall by 2.9%, mAP0.5 by 1.8% and mAP0.5-0.95 by 2.8%. In experiment 6, modifying the neck structure and C2PSA raised recall by 2.9%, mAP0.5 by 1.7%, and mAP0.5-0.95 by 2.9%, while reducing model complexity. Experiment 7 demonstrates that YOLOv11-MHS, which achieves optimization in accuracy, robustness and lightweight design through the combined effects of C3k2_MSCB, HS-FPN and C2PSA_SCSA has a 2.1% higher precision, 1.7% higher recall,

Table 1 YOLOv11-MHS tomato ripeness detection results

Group	MSCB	HS-FPN	SCSA	Precise (%)	Recall (%)	Mean average precision		GFLOPs (G)	Params	Memory size (MB)
						mAP0.5 (%)	mAP0.5-0.95 (%)			
				89.9	87.5	93.3	82.6	6.3	2,582,542	5.5
1	√			90.6	87.2	93.6	83.5	6.1	2,355,714	5.1
2		√		89.4	87.9	93.2	82.3	5.6	1,845,630	4.0
3			√	87.3	89.5	94.1	84.0	6.3	2,538,854	5.4
4	√	√		90.8	88.2	94.1	83.6	6.7	1,858,854	4.0
5	√		√	91.2	90.4	95.1	85.4	6.2	2,411,622	5.1
6		√	√	89.5	90.4	94.5	84.5	5.6	1,800,646	3.9
7	√	√	√	92.0	89.2	95.0	85.5	5.5	1,673,414	3.7

1.7% higher mAP0.5 and 2.9% higher mAP0.5-0.95 than YOLOv11n, along with a reduction of 0.8 GFLOPs, 909,128 parameters and 1.8 MB memory.

The ablation experiments show the enhanced YOLOv11-MHS model can more accurately identify tomatoes at varying ripeness stages during detection tasks. The optimized model cuts hardware resource needs and reduces storage and computational resource use.

3.4 Comparison with other algorithms

To comprehensively assess the performance of the proposed YOLOv11-MHS model in tomato ripeness detection and to ensure a fair and standardized comparison, we conducted systematic comparative experiments between our method and mainstream object detectors, including Faster-RCNN, YOLOv7, YOLOv8n, YOLOv10 and YOLOv11n as well as improved YOLOv5s introduced by Hu et al.[18] and the YOLOv8-Tomato proposed by Zheng et al.[17] All models were trained and evaluated on the same data set under identical hyperparameter settings, ensuring consistency in experimental conditions. The evaluation results are detailed in Table 2.

As presented in Table 2, the proposed YOLOv11-MHS model exhibits superior performance compared to several commonly used object detection models in terms of both detection accuracy and the model being lightweight. Specifically, the precision of YOLOv11-MHS is 2.9%, 4.5%, 5.0%, 2.1%, 5.0%, 7.7% and 3.0% higher than that of YOLOv7-tiny, YOLOv8n, YOLOv10n, YOLOv11n, Faster-RCNN, improved YOLOv5s and YOLOv8-Tomato, respectively. In terms of recall, the model achieves values that are 8.2%, 2.8%, 1.1%, 1.7%, 0.8%, 4.1% and 3.1% higher than these other models, respectively, indicating a stronger ability to identify true positives across complex ripeness conditions. For detection accuracy, YOLOv11-MHS achieves mAP0.5 values that are 8.7%, 2.2%, 2.9%, 1.7%, 14.8%, 8.3% and 2.3% higher, and mAP0.5-0.95 values that are 14.1%, 5.2%, 10.6%, 2.9%, 9.6%, 13.2% and 4.8% higher than the other models, respectively. Beyond detection accuracy, YOLOv11-MHS also has clear advantages in terms of model being lightweight: GFLOPs of YOLOv11-MHS are 7.5, 2.6, 2.7, 0.8, 129, 8.1 and 2.8 lower than the other models, respectively. The number of parameters is reduced by 4,336,888, 1,332,624, 1,021,782, 909,128, 39,679,586, 4,219,938 and 1,617,872, respectively. Correspondingly, the memory usage is 8.6 MB, 2.5 MB, 2.1 MB, 1.8 MB, 330.4 MB, 8.1MB and 2.8 MB lower, respectively. Overall, YOLOv11-MHS surpasses all comparison models in both detection performance and its lightweight design, making it a highly effective and well-suited solution for tomato ripeness detection tasks.

Figure 8 compares detection results across backlight, shadowy, fruit overlap, and branch and leaf cover, visually illustrating performance differences. Purple and blue boxes denote ripe and unripe tomatoes, respectively, and red circles denote errors. Under backlighting, only YOLOv11-MHS locate the leftmost unripe tomato, and in shadowy conditions, all models except YOLOv11-MHS missed several tomatoes. these results highlight their poor robustness to contrast and brightness variations. All models work in the fruit-overlap scenarios, but under foliage occlusion only Faster-RCNN, improved YOLOv5s and YOLOv11-MHS gave zero misses, as others cannot distinguish fruit from similar leaf textures. This indicates that the two-stage model does provide distinct

Table 2 Comparative results of eight object detection algorithm models evaluated

Model	Precise (%)	Recall (%)	Mean average precision		GFLOPs (G)	Params	Memory size (MB)
			mAP0.5 (%)	mAP0.5-0.95 (%)			
YOLOv7tiny	89.1	81.0	86.3	71.4	13	6,010,302	12.3
YOLOv8n	87.5	86.4	92.8	80.3	8.1	3,006,038	6.2
YOLOv10n	87.0	88.1	92.1	74.9	8.2	2,695,196	5.8
YOLOv11n	89.9	87.5	93.3	82.6	6.3	2,582,542	5.5
Faster-RCNN	87.0	88.4	80.2	75.9	134	41,353,000	334
Improved YOLOv5s	84.3	85.1	86.7	72.3	13.6	5,893,352	12.2
YOLOv8-tomato	89.0	86.1	92.7	80.7	8.3	3,291,286	6.8
YOLOv11-MHS	92.0	89.2	95.0	85.5	5.5	1,673,414	3.7

Fig. 8 Detection results of different algorithm models in complex environments. (a) YOLOv8n; (b) YOLOv10n; (c) YOLO11n; (d) faster-RCNN; (e) improved YOLOv5s; (f) YOLOv8-tomato; and (g) YOLOv11-MHS.

advantages in addressing severe occlusion. In future research, YOLOv11-MHS could integrate the strengths of Faster-RCNN to further enhance its performance in occluded scenarios.

The enhancements of YOLOv11-MHS compared to other models are mainly attributed to the spatial attention and channel attention mechanisms in the C2PSA_SCSA module, which enhance the ability of the model to detect tomatoes from complex backgrounds. The C3k2_MSCB, by enhancing feature extraction capabilities, using depthwise separable convolutions, and leveraging interchannel relationships, works together with the feature fusion structure of HS-FPN to improve the stability of the model under different lighting conditions and performance in tomato detection. Meanwhile, the optimization of the Neck layer by HS-FPN and the depthwise separable convolutions of C3k2_MSCB reduce the computational load and model size. YOLOv11-MHS, which combines high precision, stability and low computational cost, is very suitable for maturity detection under outdoor conditions.

4 Conclusions

In this paper, we have presented an optimized YOLOv11n-based model, YOLOv11-MHS, for tomato ripeness detection. We introduced the MSCB to create C3k2_MSCB, which leverages multiscale feature fusion to elevate detection accuracy. The neck was strengthened via a HS-FPN that refines features through selection and fusion, bolstering robustness in open-air orchard scenarios. Additionally, we integrated SCSA to develop C2PSA_SCSA, which includes SMSA for strengthening feature extraction and PCSA for improving small-target detection. The refined YOLOv11-MHS model elevated detection accuracy while achieving a lightweight design. Ablation studies confirmed the effectiveness of these improvements. Relative to YOLOv11n, it boosts precision by 2.1%, recall by 1.7%, mAP0.5 by 1.7% and mAP0.5-0.95 by 2.9%. It also reduced GFLOPs by 0.8, parameters by 909,128, memory size by 1.8 MB. When benchmarked against mainstream object detection models, YOLOv11-MHS gave better performance across all metrics and provided the best detection results in natural settings.

In the future, the model developed in this study could be integrated into devices, such as autonomous harvesting robots and orchard monitoring systems, contributing to advancing precision agriculture and reducing labor costs.

Acknowledgements

This work was financially supported by National Natural Science Foundation of China (12364011), Guangxi Science and Technology Plan, China (AD21220147, AD25069027), Liuzhou Science and Technology Program, China (2023PRJ0103, 2024AA0204A001), and Graduate Education Innovation Project, China (YCSW2024522).

Compliance with ethics guidelines

Dongyang Wang, Zhijie Fang, Man Mo, Jinchong Gan, and ZiJun Sun declare that they have no conflicts of interest or financial conflicts to disclose. This article does not contain any studies with human or animal subjects performed by any of the authors.

REFERENCES

1. Vats S, Bansal R, Rana N, Kumawat S, Bhatt V, Jadhav P, Kale V, Sathe A, Sonah H, Jugdaohsingh R, Sharma T R, Deshmukh R. Unexplored nutritive potential of tomato to combat global malnutrition. *Critical Reviews in Food Science and Nutrition*, 2022, **62**(4): 1003–1034
2. Salehi B, Sharifi-Rad R, Sharopov F, Namiesnik J, Roointan A, Kamle M, Kumar P, Martins N, Sharifi-Rad J. Beneficial effects and potential risks of tomato consumption for human health: an overview. *Nutrition*, 2019, **62**: 201–208
3. Qiu R, Song J, Du T, Kang S, Tong L, Chen R, Wu L. Response of evapotranspiration and yield to planting density of solar greenhouse grown tomato in Northwest China. *Agricultural Water Management*, 2013, **130**: 44–51
4. Yadav A, Kumar N, Upadhyay A, Sethi S, Singh A. Edible coating as postharvest management strategy for shelf-life extension of fresh tomato (*Solanum lycopersicum* L.): an overview. *Journal of Food Science*, 2022, **87**(6): 2256–2290
5. Liu Z G, Wang L J, Xi G N, Peng C H, Jiao Y Q. Present

situation and development of fruit maturity detection technology. *Agriculture and Technology*, 2020, **40**(8): 17–21 (in Chinese)

6. Dai G W, Fan J C, Yan S, Hu L. A Method for Greenhouse Tomato Recognition and Detection Using YOLOv5 Improved with CBAM and Octave Convolution. *Agricultural Information Institute of Chinese Academy of Agricultural Sciences*, 2023, Patent No.: CN202210728904.0 (in Chinese)
7. Chen Z, Chen Y, Li H, Wang P. Design and control algorithm of a motion sensing-based fruit harvesting robot. *Frontiers of Agricultural Science and Engineering*, 2025, **12**(2): 190–207
8. Zhang J, Kang N, Qu Q, Zhou L, Zhang H. Automatic fruit picking technology: a comprehensive review of research advances. *Artificial Intelligence Review*, 2024, **57**(3): 54
9. Dai G, Tian Z, Wang C, Tang Q, Chen H, Zhang Y. Lightweight vision transformer with lite-AVPSO hyperparameter optimization for agricultural disease recognition. *IEEE Internet of Things Journal*, 2025 [Early Access]
10. Dai G, Tian Z, Fan J, Sunil C K, Dewi C. DFN-PSAN: multi-level deep information feature fusion extraction network for interpretable plant disease classification. *Computers and Electronics in Agriculture*, 2024, **216**: 108481
11. Dai G, Fan J, Dewi C. ITF-WPI: image and text based cross-modal feature fusion model for wolfberry pest recognition. *Computers and Electronics in Agriculture*, 2023, **212**: 108129
12. Dai G, Fan J, Tian Z, Wang C. PPLC-Net: neural network-based plant disease identification model supported by weather data augmentation and multi-level attention mechanism. *Journal of King Saud University - Computer and Information Sciences*, 2023, **35**(5): 101555
13. Mijwil M M, Aggarwal K, Sonia S, Al-Mistarehi A H, Alomari S, Gök M, Zein Alaabdin A M, Abdulrhman S H. Has the future started? The current growth of artificial intelligence, machine learning, and deep learning. *Iraqi Journal for Computer Science and Mathematics*, 2022, **3**(1): 115–123
14. Kang H, Chen C. Fast implementation of real-time fruit detection in apple orchards using deep learning. *Computers and Electronics in Agriculture*, 2020, **168**: 105108
15. Chen S, Xiong J, Jiao J, Xie Z, Huo Z, Hu W. *Citrus* fruits maturity detection in natural environments based on convolutional neural networks and visual saliency map. *Precision Agriculture*, 2022, **23**(5): 1515–1531
16. Ma L, He Z, Zhu Y, Jia L, Wang Y, Ding X, Cui Y. A method of grasping detection for kiwifruit harvesting robot based on deep learning. *Agronomy*, 2022, **12**(12): 3096
17. Zheng S, Jia X, He M, Zheng Z, Lin T, Weng W. Tomato recognition method based on the YOLOv8-tomato model in complex greenhouse environments. *Agronomy*, 2024, **14**(8): 1764
18. Hu Y F, Zhao X L, Li P J, Zhao C Y, Chen G M. Tomato fruit detection in natural environment based on improved YOLOv5. *Journal of Chinese Agricultural Mechanization*, 2023, **44**(10): 231–237 (in Chinese)
19. Wang Y, Rong Q, Hu C. Ripe tomato detection algorithm based on improved YOLOv9. *Plants*, 2024, **13**(22): 3253
20. Ma P W, Zhou J. Detecting grape ripeness in complex environments using improved YOLOv7. *Transactions of the Chinese Society of Agricultural Engineering*, 2025, **41**(3): 171–178 (in Chinese)
21. Xie Z, Chen R, Lin C, Zeng J. A lightweight real-time method for strawberry ripeness detection based on improved YOLO. *Social Science Research Network (SSRN)*, 2023 [Preprint]
22. Majdudin R A H. Pineapple ripeness detection using YOLOv8 algorithm. *Publication of the International Journal and Academic Research*, 2025, **1**(1): 50–54
23. Parvathi S, Tamil Selvi S. Detection of maturity stages of coconuts in complex background using Faster R-CNN model. *Biosystems Engineering*, 2021, **202**: 119–132
24. Rahman M M, Munir M, Marculescu R. EMCAD: Efficient Multi-scale Convolutional Attention Decoding for Medical Image Segmentation. In: 2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). Seattle, WA, USA: *IEEE*, 2024, 11769–11779
25. Chen Y, Zhang C, Chen B, Huang Y, Sun Y, Wang C, Fu X, Dai Y, Qin F, Peng Y, Gao Y. Accurate leukocyte detection based on deformable-DETR and multi-level feature fusion for aiding diagnosis of blood diseases. *Computers in Biology and Medicine*, 2024, **170**: 107917
26. Si Y, Xu H, Zhu X, Zhang W, Dong Y, Chen Y, Li H. SCSA: exploring the synergistic effects between spatial and channel attention. *Neurocomputing*, 2025, **634**: 129866
27. tomato. tomato Dataset (Open Source Dataset). *Roboflow*, 2024. Accessed at Roboflow website on June 3, 2025
28. Baijiahao B. Analysis of Tomato Cultivation: Key Factors in Timing, Yield, and Prospects. *Baidu*, 2025. Accessed at Baidu Baijiahao website on August 3, 2025 (in Chinese)
29. Centre for Indian Knowledge Systems (Chennai). HARVESTING (Organic Farming: Organic Farming Practices). *Tamil Nadu Agricultural University*, 2014. Accessed at Tamil Nadu Agricultural University agriTech website on August 3, 2025
30. Amna M W, Guiqiang L I, Muhammad Zuhaib Akram M F. Machine vision-based automatic fruit quality detection and grading. *Frontiers of Agricultural Science and Engineering*, 2025, **12**(2): 274–287