TOMATO MATURITY DETECTION BASED ON IMPROVED YOLOv8n
/
基于改进 YOLOv8n 的番茄果实成熟度检测

JunMao LI, ZiLu HUANG, LingQi XIA, Hao SUN, HongBo WANG*
College of Mechanical and Electronic Engineering, Inner Mongolia Agricultural University, Hohhot / China
Tel: +86 13739981395; E-mail: wanghb@imau.edu.cn
DOI: https://doi.org/10.35633/inmateh-75-53

Keywords: *tomato detection; YOLOv8n; fruit maturity detection; EMA attention mechanism; C2f-Faster module*

ABSTRACT
The detection of tomatoes for automatic picking is challenging due to the dense distribution of fruit and severe occlusions. To address this, a dataset is developed using tomato images captured in a greenhouse environment, and an enhanced model for tomato fruit maturity detection based on YOLOv8n is proposed, which incorporates the EMA attention mechanism and the C2f-Faster module for multi-scale feature fusion. These additions not only improve detection accuracy but also enhance detection speed, thereby boosting the model's robustness and generalization ability. Experimental results demonstrate that the proposed ECF-YOLOv8n model achieves detection accuracies of 93.8%, 94.7%, 92.5% and 94.1% for immature, nearly mature, ripe tomatoes and mean average precision in a greenhouse setting, respectively. The model's size is 4.7 MB, with GFLOPs of 6.5G. Compared to advanced models like RT-DETR, YOLOv5, YOLOv7 and YOLOV11, the ECF-YOLOv8n model outperforms them in both detection accuracy and speed. This work provides valuable insights for the research, development and optimization of tomato picking robots.

摘要
针对目前番茄自动化采摘目标检测中因果实密集、遮挡严重等导致目标检测难度大的问题，本研究基于温室大棚环境下的番茄图像，构建了数据集，提出了一种基于 YOLOv8n 的番茄果实成熟度检测的改进模型，并添加引入了 EMA 注意力机制和 C2f-Faster 模块，以实现多尺度特征融合，在保证检测精度较高的情况下，有效提高了番茄果实检测速度，从而进一步提高了模型的鲁棒性和泛化能力。试验结果表明：提出的 ECF-YOLOv8n 模型对温室大棚环境下未成熟、将要成熟、成熟番茄检测精度和均值平均精度分别为：93.8%、94.7%、92.5%和 94.1%，模型大小为 4.7 MB，GFLOPs 为 6.5G，与 RT-DETR、YOLOv5、YOLOv7、YOLOV11 等先进模型的比较，该模型实现了较高的检测精度和更快的检测速度，本研究可为番茄采摘机器人的研发和优化提供重要参考。

INTRODUCTION
Tomato is one of the important economic crops in greenhouses. In recent years, the area of greenhouse tomato cultivation has continued to expand. However, tomato harvesting is still primarily carried out by humans, which is inefficient and costly. In addition, since the harvesting window for tomatoes is short, failing to pick the ripe fruits in time directly affects both fruit quality and economic benefits *(Malik et al., 2018; Lawal et al., 2021)*. To achieve efficient and rapid automated tomato picking, accurate target detection is crucial. Target detection technology provides precise information for mechanized picking, enabling the automation of the harvesting process. Therefore, enhancing the accuracy of target detection is key to improving picking efficiency and reducing costs *(Tsai et al., 2022; Li R et al., 2023; Miao, et al., 2023)*.

In recent years, convolutional neural networks (CNNs) based on deep learning have become a major research focus and have been widely applied to the identification of greenhouse tomatoes *(Gao et al., 2022; Zeng et al., 2024)*. Target detection algorithms are generally categorized into two main types: one-stage and two-stage. Typical representatives of one-stage object detection algorithm include the YOLO series. The typical representative of the two-stage object detection algorithm is the RCNN series, including Fast R-CNN, Faster R-CNN and Mask R-CNN *(Bai et al., 2024; Yin et al., 2024; Babu et al., 2024)*.

JunMao LI, M.S. Stud.; ZiLu HUANG, M.S. Stud.; LingQi XIA, M.S. Stud.; Hao SUN, Stud.;
HongBo Wang, Professor, Correspondent author

This algorithm first generates a series of candidate boxes for samples and then classifies the samples through a convolutional neural network. Compared with the one-stage algorithm, the two-stage algorithm has a slower detection speed, and a higher algorithm complexity.

In the field of fruit and vegetable maturity detection, numerous experts and scholars both domestically and internationally have conducted relevant research. In terms of one-stage algorithm, *Fengjun et al.*, *(2024)*, addressed the issue of occlusion of Camellia oleifera fruits in natural environments by improving the original YOLOv7 model. They proposed a maturity detection method for Camellia oleifera fruits, providing a theoretical basis for the intelligent harvesting of these fruits. To improve the accuracy of surface defect detection in pinewood while maintaining detection speed, *Jiwen et al.*, *(2024)*, proposed an improved RT-DETR model, RIC-DETR. Experimental results demonstrated that the RIC-DETR model achieved an accuracy of 95.4%, offering technical support for surface defect detection in pinewood. *Zheng et al. (2022)* constructed a new backbone network, R-CSPDarknet53, based on YOLOv4 by integrating a residual neural network to establish skip connections between the front and back layers, thereby preventing the loss of low-dimensional small target features. In addition, by replacing the maximum pool in the original SPP network with the deep separable convolution model, C-SPP is proposed to realize feature information reuse and multi-scale fusion. On this basis, a tomato detection model RC-YOLOv4 is constructed, which improves the detection accuracy of tomato in natural environment. The test results show that the tomato detection accuracy and recall rate of RC-YOLOv4 model in natural environment are 88% and 89% respectively, the average detection accuracy is 94.44%. *Appe et al.*, *(2023)*, proposed an improved YOLOv5 tomato detection algorithm. By adding CBAM convolutional attention mechanism to the YOLOv5 model, feature extraction and target recognition were carried out to improve the accuracy of the model. Non-maximal suppression and distance union ratio (DIoU) were APPLIED to enhance the recognition of overlapping objects in the image. The results showed that the average accuracy of the CAM-YOLO algorithm for the detection of overlap and small tomatoes was 88.1%.

Li P. et al. (2023), based on the requirements of the tomato maturity grading task, adopted the MHSA attention mechanism to improve the YOLOv8 backbone, enhancing the network's ability to extract diverse features. The Precision, Recall, F1-score, and mAP50 of the tomato fruit maturity grading model constructed based on MHSA-YOLOv8 were 0.806, 0.807, 0.806, and 0.864. *Solimani et al., (2024),* proposed a new data balancing method in order to overcome the problem of data imbalance. A squeezing and exciting (SE) block attention module is integrated into the head structure of YOLOv8 model, which significantly improves the algorithm's ability to detect objects of different sizes in complex environments, and can effectively detect flowers and fruits in tomato plants.

In terms of two-stage algorithm, *Gao et al.*, *(2020)*, addressed the issue of occlusion in apples during the harvest period by applying the Faster R-CNN model for detecting occluded apples. Experimental results showed that the model achieved an average detection accuracy of 80%-90% for occluded apples. *Chen et al.*, *(2022)*, integrated Gabor features into Faster R-CNN and proposed a two-stage training method based on a genetic algorithm and backpropagation to train a new Faster GG-R-CNN model, achieving an average precision of 94.57%. *Seo et al.*, *(2021)*, developed a real-time robotic detection system based on Faster R-CNN, utilizing hue values to establish an image-based ripeness standard for tomatoes, with a recognition accuracy of 90.2%. *Fang et al., (2024),* proposed a multi-target identification and localization method for tomato plants based on the VGG16-UNet model. The average intersection and pixel accuracies of the VGG16-UNet model after introducing the pretrained weights were 85.33% and 92.47%, respectively, which were 5.02% and 4.08% higher than those of the VGG16-UNet without pretrained weights, achieving the identification of main branches, side branches, and axillary bud regions.

In a greenhouse environment, factors such as lighting, occlusion, and the density of plants can complicate the accurate identification of tomato ripeness *(Huiqin et al., 2024)*, leading to low detection efficiency. To improve the detection accuracy and speed of tomato ripening in greenhouse, this study proposes an enhanced YOLOv8n object detection algorithm. By incorporating the EMA attention mechanism, the model reduces sensitivity to noise and outliers, while the C2f-Faster module enables multi-scale feature fusion, thus improving both detection accuracy and speed for tomatoes. To account for the complexity of greenhouse tomato scenes, images of tomatoes are captured under various weather conditions, lighting angles, and shooting perspectives, ensuring the dataset's richness and diversity. Additionally, preprocessing and image augmentation techniques are applied to enhance dataset quality, making it better suited for tomato fruit detection in greenhouse environments. This research provides valuable target information for tomato-picking robots and offers a theoretical foundation for automated harvesting.

MATERIALS AND METHODS

DATA SAMPLE COLLECTION AND DATASET CREATION

Data Sample Collection

The tomato image data were collected from a tomato picking garden located in the suburbs of Hohhot, where standardized cultivation techniques are applied. To ensure the diversity of the dataset, enhance the model's robustness, and improve its generalization ability, images of tomatoes were captured at different times of day, across various stages of ripeness, from different angles, at varying distances, and under different lighting conditions *(Wang et al., 2024)*. A total of 2,080 images, each with a resolution of 640×640, were selected for the dataset, as shown in Figure 1.

Fig. 1 - Images of tomatoes in a greenhouse in different scenes

Dataset Preparation

The creation of the dataset primarily involved two key processes: image annotation and dataset categorization. The LabelImg annotation tool was used for manual labelling of the image data, following the YOLO dataset annotation format. According to the national standard GH/T1193-2021, the maturity of tomatoes can be divided into unripe stage, green ripe stage, colour change stage, early red ripe stage, mid-red ripe stage and late red ripe stage. Specifically, tomatoes in the mid-red ripe stage and late red ripe stage have a red surface coverage of 40%-60% and 70%-100%, respectively. In the greenhouse, only mid-red and late-red ripe tomatoes are harvested *(Zhao, 2024)*.

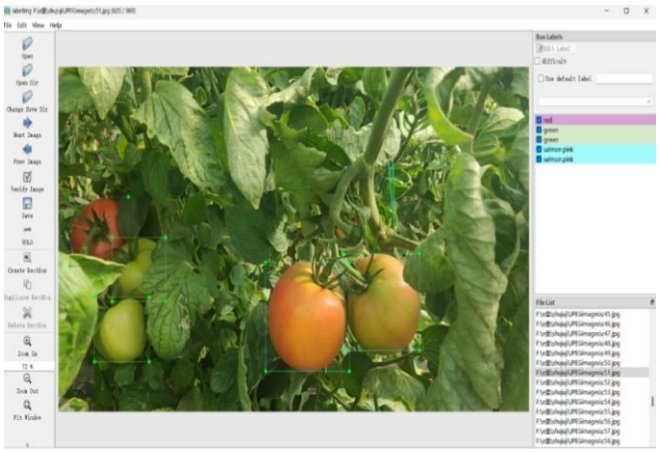


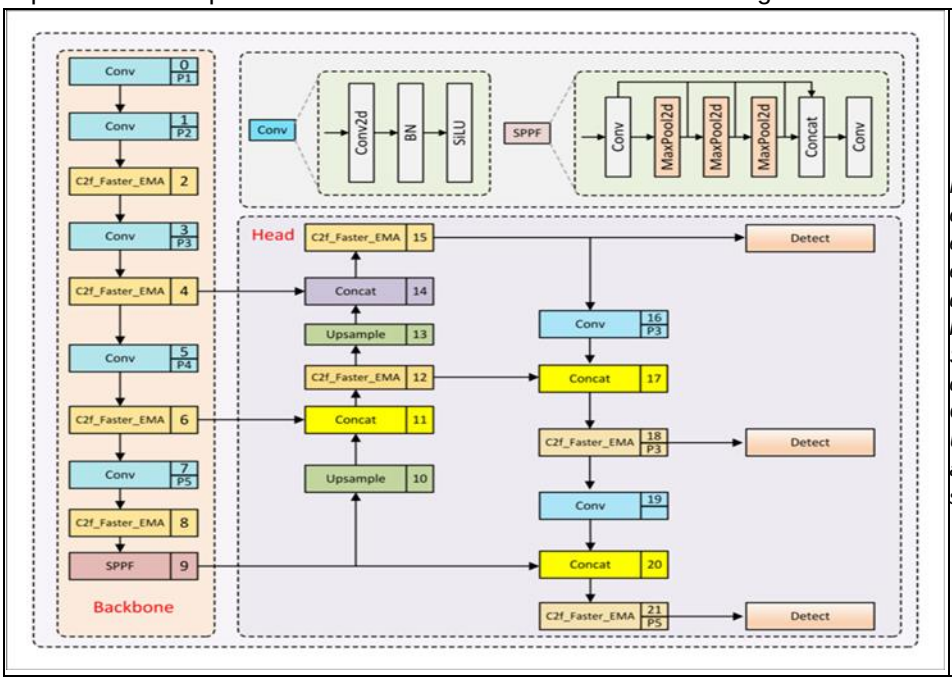
Fig. 2 - Tomato image annotation

In this paper, the labelled targets, i.e. the tomato fruits to be detected, are divided into three categories: Tomato fruits in the unripe, green-ripe, colour-changing, and early red-ripe stages, i.e. tomatoes with less than 40% red surface coverage, are classified as unripe and labelled as green; mid-ripe tomatoes with 40% to 60% red colouring on the fruit surface are classified as almost ripe and are labelled as salmon pink; late-ripe tomatoes with 70% to 100% red colouring on the fruit surface are classified as mature and labelled as red. The image annotations are illustrated in Figure 2. Upon completion of the annotation process, a corresponding text file (*.txt) for each image was generated, linking the image name with the txt file name. The labelled dataset was then split into training, validation, and test sets in a 7:1:2 ratio, with 1456 images for training, 208 images for validation, and 416 images for testing.

FRUIT MATURITY DETECTION MODEL OF TOMATO

YOLOv8 algorithm is the Yolo series target detection algorithm launched by Ultralytics. It is an upgrade based on the historical version of the Yolo series. The network composition of YOLOv8 mainly includes four parts: Input, Backbone, Neck and Head. Backbone is the network part used to extract image features in YOLOv8. It uses a series of convolution and deconvolution layers to extract features, and also uses residual connections and bottleneck structures to reduce the size of the network and improve performance. The neck part plays a role in feature fusion in YOLOv8. It uses multi-scale feature fusion technology to fuse feature maps from different stages of the backbone to enhance feature representation capabilities. The head part is responsible for the final target detection and classification tasks.

YOLOv8 has five different structures, namely YOLOv8m, YOLOv8l, YOLOv8x, YOLOv8n, and YOLOv8s. These models differ only in depth and width. The basic structure of these models is four parts. In order to meet the requirements of lightweight and real-time detection, while ensuring high detection accuracy and detection speed, YOLOv8n which has a relatively low complexity is chosen as the base model. It can achieve faster recognition speed and smaller storage occupancy while ensuring high detection accuracy, which is conducive to deployment on mobile devices *(Hussain et al., 2023)*. Based on YOLOv8n, an improved ECF-YOLOv8n network model structure is proposed, and the EMA attention mechanism is introduced. By reconstructing some channels into batch dimensions and grouping the channel dimensions into multiple sub-features, the EMA attention mechanism can reduce information loss while keeping the tensor size unchanged, and enhance the model's ability to capture spatial semantic features. The C2f-Faster-EMA module is also responsible for fusing feature maps of different scales to generate more representative feature representations. This feature fusion process may be achieved through upsampling, downsampling, splicing and other operations to ensure that the model can fully utilize multi-scale information to improve detection performance. While keeping the YOLOv8n model lightweight, the detection performance and speed of the model are improved. The improved ECF-YOLOv8n structure is shown in Figure 3.



Note: Conv is a convolution operation; C2f-Faster-EMA is efficient multi-scale feature extraction. Conv2d is a two-dimensional convolution; BN is batch normalized; SiLu is the Sigmoid weighted linear combinatorial activation function; Concat is a splicing operation; UPSample for upsampling; SPFF is a fast space pyramid pooling structure

Fig. 3 - Structural diagram of the improved YOLOv8n

EMA Attention Mechanism

EMA attention mechanism is a new type of efficient multi-scale attention method, which focuses on retaining information on each channel and reducing the amount of computation, as shown in Figure 4.

EMA is an attention weight descriptor that uses three parallel paths to extract grouped feature maps. Two of the parallel routes are located in the 1×1 branch, and the third parallel route is located in the 3×3 branch. In order to reduce the amount of computation and obtain the dependencies between channels at the same time, cross-channel information interaction is established in the channel direction.

In this structure, output represents the output plane of the input features, input represents the input plane of the input features, and k represents the kernel size. Accordingly, the G group is reshaped to the batch dimension and the input tensor is redefined as C//G×H×W. The two encoded features are connected in the height direction of the image and share the same 1×1 convolution without dimensionality reduction 1×1 branch *(Xu et al., 2024)*. After decomposing the output of the 1×1 convolution into two vectors, two nonlinear sigmoid functions are used to fit the 2D Binomial distribution on the linear convolution. In order to realize the different cross-channel interaction features between the two parallel paths in the 1×1 branch, the attention maps of the two channels are aggregated within each group by a simple multiplication. On the other hand, the 3×3 branch captures the local cross-channel interaction through 3×3 convolution to expand the feature space. In this way, EMA not only encodes inter-channel information to adjust the importance of different channels, but also embeds the precise spatial structure information into the channel *(Yang et al., 2024)*.

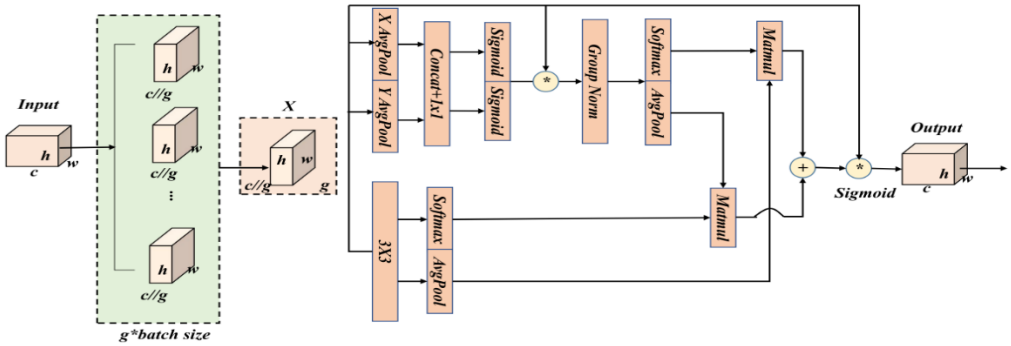


Fig. 4 - EMA module

C2f module

C2f module consists of a convolution block, which receives the input feature map and generates an intermediate feature map. The C2f module structure diagram is shown in Figure 5. The generated intermediate feature map is split into two parts, one part is directly passed to the final “Concat block”, and the other part is passed to multiple “Bottleneck blocks” for further processing. The feature map input to the "Botleneck block" is processed through a series of convolution, normalization and activation operations, and the final feature map is concatenated with the directly passed feature map in the "Concat block". In the C2f module, the number of “Bottleneck modules” is defined by the “depth multiple” parameter of the model, that is, the depth and computational complexity of the module can be adjusted according to the needs. The concatenated feature maps are input into a final convolutional block for further processing to generate the final output feature map.

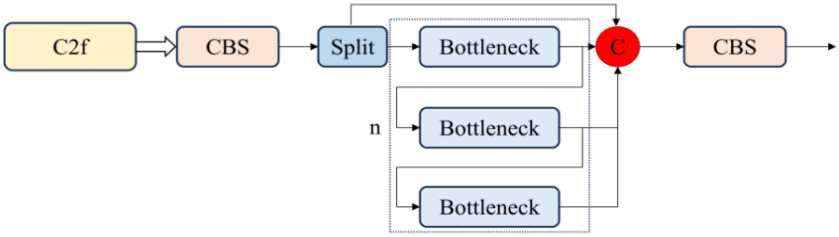


Fig. 5 - C2f module

C2f-Faster Module

In the target detection task, due to the relatively simple structure of the C2f module and the lack of a dedicated acceleration mechanism, it exhibits a low detection speed when processing large-scale data sets or performing real-time detection. In order to solve this performance bottleneck, a new neural network, FasterNet, was introduced. FasterNet has shown significant advantages in achieving fast target detection due to its excellent running speed and optimized design.

There is a new partial convolution (PConv) in the FasterNet module, as shown in Figure 6 (c). The core function of partial convolution (PConv) lies in its flexibility and adaptability to data missing. Compared with traditional convolution, partial convolution does not mechanically apply the same convolution kernel to all parts of the input data. Instead, it dynamically determines the scope of the convolution kernel based on the validity of the data, that is, whether the data points are missing or damaged.

When partial convolution (PConv) processes a convolution window, it first checks the data points within the window. For valid, non-missing data points, PConv applies the convolution kernel like a regular convolution operation. However, for missing or invalid data points, PConv will choose to ignore them and not include them in the convolution calculation. This flexibility means that the actual area of action of the convolution kernel may be different in each convolution window. This depends entirely on the completeness and distribution of the data in the window. In this way, partial convolution not only improves the robustness to missing data, but also more effectively extracts and utilizes the remaining valid information. By reducing redundant computation and memory access at the same time, spatial features can be extracted more efficiently. Each FasterNet (FasterBlock) module has a PConv layer followed by two Conv1×1 layers. Together, they are shown as an inverted residual block, where the intermediate layers have an expanded number of channels and shortcut connections are placed to reuse input features.

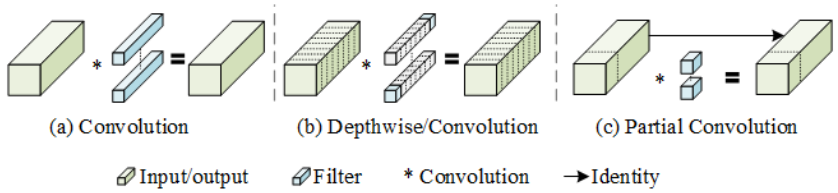


Fig. 6 - Three different convolution structures

The intermediate feature map generated by the fusion of the C2f module and FasterNet is split into two parts, one part is directly passed to the final Concat block, and the other part is passed to multiple FasterBlock blocks for further processing. The C2f-Faster module structure diagram is shown in Figure 7. The feature map input to the FasterBlock is further processed through a series of partial convolution, normalization and activation operations. At this time, the FasterBlock is more efficient and faster than the Bottleneck due to the existence of the partial convolution layer. The final feature map will be concatenated with the directly transmitted feature map in the Concat. The concatenated feature map will be input to a final convolution block for further processing to generate the final output feature map.

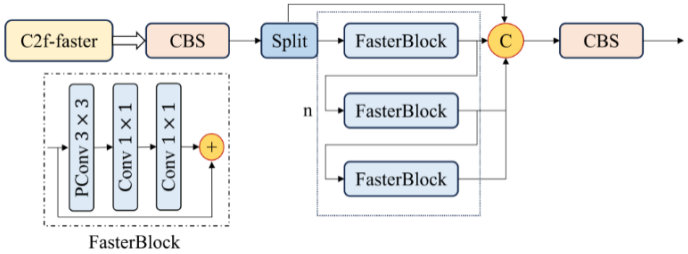


Fig. 7 - C2f-Faster module

C2f-Faster-EMA Module

Although C2f-Faster speeds up the detection speed, some convolution operations will cause some information loss, resulting in a decrease in detection accuracy, which cannot meet the requirements of tomato maturity detection in a greenhouse. Therefore, the EMA (Efficient Multi-scale Attention) attention mechanism is integrated in the C2f-Faster module, forming the C2f-Faster-EMA module, as shown in Figure 8.

The C2f-Faster-EMA module enhances the model's focus on key features, allowing it to more accurately identify target objects during detection, thereby improving overall detection accuracy. At the same time, the acceleration characteristics of FasterNet enable the C2f-Faster-EMA module to significantly reduce computational complexity and time consumption during target detection, speeding up the detection process. The FasterEMA module consists of a partial convolution layer (PConv) followed by a Sequential module comprising a multi-layer perceptron (MLP). This Sequential module includes a CBS module and a convolution layer (Conv2d). To further enhance the model's generalization ability and effectively prevent overfitting, DropPath regularization strategy is introduced. Specifically, DropPath randomly selects a subset of paths within each feature map and sets the weights of these paths to zero. This reduces the number of effective paths, thereby decreasing the model's parameter count and improving its robustness.

The Droppath operation can be applied to each feature map of every convolutional layer, with the pruning probability dynamically adjusted during training to control the extent of pruning. Finally, the EMA attention mechanism, incorporated into the C2f-Faster-EMA, enhances the feature fusion capability of the C2f module. By introducing a more complex network structure and attention mechanism, the model is able to learn more rich and comprehensive feature representations. These representations not only improve detection accuracy but also contribute to the model's robustness in more complex scenarios.

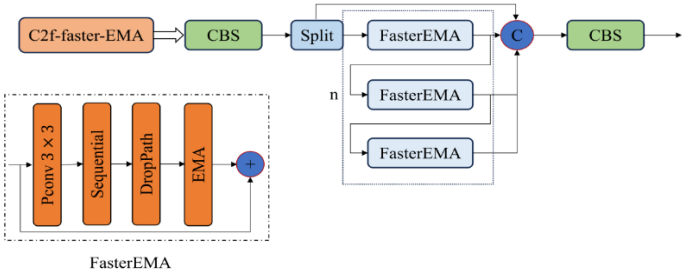


Fig. 8 - C2f-Faster-EMA module

Test environment

The operating system used for trial and training is Windows 11, the CPU is 12th Gen Intel(R) Core(TM) i5-12500H 3.10 GHz, the GPU is NVIDIA GeForce RTX 3060 Laptop GPU, and the running memory is 16G. CUDA version is 12.0, and is implemented using Python 3.10.14 under the PyTorch2.1.0 deep learning framework.

Evaluation indicators

Seven indicators were used to evaluate the maturity detection model of tomato fruit, namely precision (P), recall (R), average precision (AP), and mean average precision (mAP), model parameters, detection speed and memory usage. The calculation formulas of P, R, AP and mAP are as follows.

$$P = \frac{T_P}{T_P + F_P} \times 100\% \quad (1)$$

$$R = \frac{T_P}{T_P + F_N} \times 100\% \quad (2)$$

$$AP = \int_0^1 P(R)dR \quad (3)$$

$$mAP = \frac{\sum_{i=1}^{m} AP(n)}{3} \quad (4)$$

$$t = \frac{t_N}{N} \quad (5)$$

where, T_P represents the number of positive samples predicted as positive samples, that is, the number of correctly predicted tomato ripening levels; F_P represents the number of negative samples predicted as positive samples, that is, the number of falsely predicted tomato ripening levels; F_N represents the number of negative samples predicted as negative samples, that is, the number of tomato ripening level is incorrectly predicted; AP represents the $P(R)$ curve made by using the Recall value as the X axis and the Precision value as the Y axis. The area of the measurement is the accuracy of identification of a certain category; mAP represents the average value of each category of AP, and measures the average quality of all categories.

RESULTS

IMPROVED TEST RESULTS OF YOLOV8N MODEL

In order to verify the effect of the improved YOLOv8n model, the accuracy P of the improved YOLOv8n model reached 93.7%, the recall R was 83.7%, and the average accuracy mean mAP was 94.1% for the 416 tomato fruit images divided into the test set. Some of the detection results are shown in Figure 9.

Fig. 9 - Improved YOLOv8n model detection results

COMPARISON OF IMPROVED YOLOV8N ABLATION EXPERIMENT PERFORMANCE

In order to better analyse the detection performance of the improved YOLOv8n model on tomato fruit maturity, ablation test was performed using YOLOv8n as the basic model to verify the optimization effect of each improved module. The optimization effect of each improvement point is evaluated using precision (P), recall (R), mean average precision (mAP), parameter quantity, floating point operations per second (FLOPs) and memory usage. The results of the ablation test are shown in Table 1.

Table 1

Ablation test results

Model	Green mature (P)/%	Colour change (P)/%	Red mature (P)/%	MAP50 /%	Parameter quantity /M	FLOPs /G	Memory usage /MB
YOLOv8n	94.5	89.6	91.9	91.9	3.0	8.1	5.6
v8n-C2f-Faster	92.2	90.5	95	92.9	2.3	6.3	4.6
v8n-C2f-Faster-CGLU	93.2	92.7	87.1	93.2	2.2	6.2	4.5
v8n-C2f-Faster-EMA	93.8	94.7	92.5	94.1	2.3	6.5	4.7

The analysis results show that the proposed ECF-YOLOv8n model with the C2f-Faster-EMA module has significantly reduced the parameters, floating point operations amount and memory usage of the model compared with the YOLOv8n basic model, which improves the computing and storage efficiency. The detection precision (P) has also been increased, with the average of the mean average precision (mAP) increased by 2.2 percentage points, and the parameter quantity, floating point operations per second (FLOPs), and memory usage have been reduced by 23.33%, 19.75%, and 21.52%, respectively. This shows that the ECF-YOLOv8n model is maintaining high detecting rate, the model is lightweighted. Although the proposed ECF-YOLOv8n model with fusion C2f-Faster-EMA module has slightly increased the mean average precision (mAP) compared with the model with fusion C2f-Faster and C2f-Faster-CGLU. The average value has increased by 1.3 and 0.9 percentage points respectively, with a valuable increase, which can achieve accurate and efficient identification of tomato fruits by picking robots in greenhouse, and is more conducive to the picking of tomato picking robots.

COMPARATIVE TEST RESULTS OF IMPROVING YOLOV8N MODEL

In order to evaluate the detection effect of the ECF-YOLOv8n model proposed in this paper on tomato fruit maturity, three algorithms, YOLOv5, YOLOv7, YOLOv11, and RTDETR, were selected for performance comparison under the premise of consistent experimental conditions. The comparison results are shown in Table 2.

Table 2

Comparison results of different models

Model	Precision (P)/ %	Recall (R) / %	Mean average precision (mAP) / %	Frames per second (FPS)	Parameter quantity / M	FLOPs / G	Memory usage / MB
RT-DETR	87.5	82.1	87.6	87.0	19.9	56.9	40.5
YOLOv5	90.0	85.0	88.7	263.6	7.1	15.8	14.4
YOLOv7	91.5	87.2	94.3	145.8	37.9	104.1	73.8
YOLOv11	90.1	87.9	93.3	107.5	2.58	6.3	4.6
Ecf-YOLOv8n	93.7	83.7	94.1	335.1	2.3	6.5	4.7

As shown in Table 2, the ECF-YOLOv8n model achieved a precision (P) of 93.7%, a recall (R) of 83.7%, and a mean average precision (mAP) of 94.1%. Compared to other detection models, the ECF-YOLOv8n model demonstrates improved performance in greenhouse tomato maturity detection. Compared with the mean average precision (mAP), the RT-DETR, YOLOv5 and YOLOv11 models were improved by 6.5%, 5.4% and 0.8%. respectively, indicating that ECF-YOLOv8n has higher precision in the detection of tomato maturity in greenhouses The ECF-YOLOv8n model has decreased by 0.2% compared with the YOLOv7 model in terms of the mean average precision (mAP), but the parameter quantity and memory usage of ECF-YOLOv8n are much smaller than that of YOLOv7, and the detection rate is faster.

The average frame of ECF-YOLOv8n reaches 335.1 frames/s, which is far higher than YOLOv7, showing that it has better efficiency and better real-time detection capabilities. Overall, ECF-YOLOv8n has shown balanced and excellent performance in detection performance, resource use and detection rate, and has better effect on the detection of tomato maturity in greenhouses.

Test verification

Owing to the constraints of test conditions, time, and other factors, a laboratory-based experiment was conducted to evaluate the detection performance of the enhanced ECF-YOLOv8n model for tomato maturity recognition. The results indicated that the average recognition accuracy of the improved ECF-YOLOv8n model for tomatoes reached 91%, which satisfies the requirements for greenhouse tomato harvesting. Future testing will be conducted in real-world greenhouse environments to improve the model's adaptability in complex scenarios, thereby better aligning it with the requirements of intelligent agriculture.

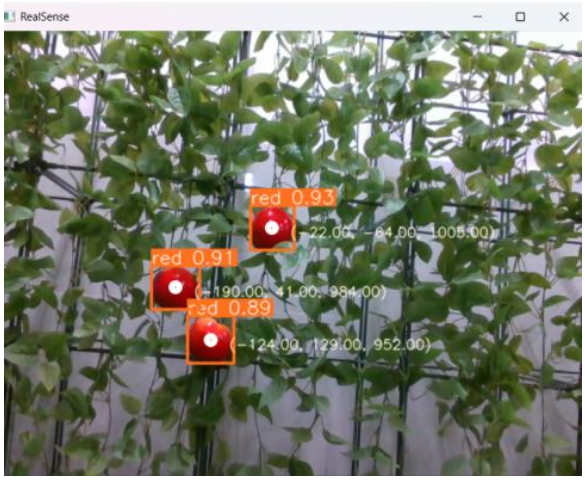


Fig. 10 - Tomato ripeness identification results

CONCLUSIONS

This paper improves the tomato maturity model based on the Yolov8n model, and realizes rapid and accurate detection of the tomato maturity in greenhouses. The main conclusions are as follows:

1) This paper proposes an ECF-YOLOv8n model based on the Yolov8n model. The precision (P) of this model for detecting the maturity of tomato is 93.7%, the recall (R) is 83.7%, and the mean average precision (mAP) is 94.1%. Through the ablation test results, it can be seen that the improved ECF-YOLOv8n model has a mean average precision (mAP) increase of 2.2% compared with the original YOLOv8n model, and has decreased by 23.33%, 19.75%, and 21.52% respectively in terms of parameter quantity, floating point operations per second (FLOPs), and memory usage, respectively. This indicates that the method proposed in this paper, which integrates the EMA attention mechanism and introduces the C2f-Faster module for multi-scale feature fusion, improves the speed of tomato maturity detection while maintaining high detection precision. It enables fast and accurate assessment of tomato maturity in greenhouse.

2) The C2f-Faster-EMA module is introduced into the backbone and head parts of the Yolov8n model to improve the network feature extraction capability. Compared with mainstream models such as RT-DETR YOLOv5 and YOLOv11 models, the mean average precision (mAP) is increased by 6.5%, 5.4% and 0.8%. respectively, and the experimental results show that the improved ECF-YOLOv8n model has a fast detection speed and high precision, which basically meets the real-time and efficient work of picking robots, and provides a theoretical basis for tomato picking technology.

3) In the laboratory environment, the detection performance of the enhanced ECF-YOLOv8n model for tomato ripeness identification was evaluated. The results demonstrated that the improved model achieved an average identification accuracy of 91%, meeting the requirements for greenhouse tomato harvesting.

ACKNOWLEDGEMENT

This research was supported by the Inner Mongolia Autonomous Region Science and Technology Innovation Guidance Project (Kcj1-202205) and Inner Mongolia Autonomous Region Science and Technology Plan Project (2022YFDZ0022, 2022YFDZ0032).

REFERENCES

[1] Appe, S. N., Arulselvi, G., Balaji, G. N. (2023). CAM-YOLO: tomato detection and classification based on improved YOLOv5 using combining attention mechanism. *PeerJ Computer Science,* 9, e1463.

[2] Bai, T., Luo, J., Zhou, S., Lu Y., Wang Y. (2024). Vehicle-type recognition method for images based on improved faster R-CNN model. *Sensors,* 24(8), 2650.

[3] Babu, P., Habelalmateen, M. I., Srikanteswara, R., Reddy, R. A., Purushotham, N. (2024). Wafer Surface Semiconductor Defect Classification Using Convolution Neural Network Based Improved Faster R-CNN. *In 2024 Second International Conference on Data Science and Information System (ICDSIS)* (pp. 1-4). IEEE.

[4] Chen, M., Yu, L., Zhi, C., Sun, R., Zhu, S., Gao, Z., Zhang, Y. (2022). Improved faster R-CNN for fabric defect detection based on Gabor filter with Genetic Algorithm optimization. *Computers in Industry*, *134*, 103551.

[5] Fenghun, CHEN., Chuang, CHEN., Xueyan, ZHU., Deyu, SHEN., Xinwei ZHANG. (2024), Detection of Camellia oleifera fruit maturity based on improved YOLOv7 (基于改进 YOLOv7 的油茶果实成熟度检测). *Transactions of the Chinese Society of Agricultural Engineering,* 40(5), 177-186.

[6] Gao, F., Fu, L., Zhang, X., Majeed, Y., Li, R., Karkee, M., Zhang, Q. (2020). Multi-class fruit-on-plant detection for apple in SNAP system using Faster R-CNN. *Computers and Electronics in Agriculture,* 176, 105634.

[7] Gao, G., Wang, S., Shuai, C., Zhang, Z., Zhang, S., Feng, Y. (2022). Recognition and Detection of Greenhouse Tomatoes in Complex Environment. *Traitement du Signal,* 39(1).

[8] Huiqin, Li., Zhaoming, Song, Cunxiang, Liu, Yatao, Xiao. (2024), Improvement of tomato fruit detection model based on YOLOv8n (基于 YOLOv8n 的番茄果实检测模型改进), Journal of Henan Agricultural University, doi:10.16445/j.cnki.1000-2340.20240511.002.

[9] Hussain, M. (2023). YOLO-v1 to YOLO-v8, the rise of YOLO and its complementary nature toward digital manufacturing and industrial defect detection. *Machines*, *11*(7), 677.

[10] Jiwen, Hu, Guoliang, Zhang, Mingzhe, Shen, Wenhao Li. (2024), Detecting surface defects of pine wood using an improved RT-DETR model (面向松木表面缺陷检测的改进 RT-DETR 模型). *Transactions of the Chinese Society of Agricultural Engineering*, 40(7), 210-218.

[11] Lawal, M. O. (2021). Tomato detection based on modified YOLOv3 framework. *Scientific Reports,* 11(1), 1447.

[12] Li, P., Zheng, J., Li, P., Long, H., Li, M., Gao, L. (2023). Tomato maturity detection and counting model based on MHSA-YOLOv8. *Sensors,* 23(15), 6701.

[13] Li, R., Ji, Z., Hu, S., Huang, X., Yang, J., Li, W. (2023). Tomato maturity recognition model based on improved yolov5 in greenhouse. *Agronomy*, *13*(2), 603.

[14] Li, X., Fang, J., Zhao, Y. (2024). A Multi-Target Identification and Positioning System Method for Tomato Plants Based on VGG16-UNet Model. *Applied Sciences*, *14*(7), 2804.

[15] Malik, M. H., Zhang, T., Li, H., Zhang, M., Shabbir, S., Saeed, A. (2018). Mature tomato fruit detection algorithm based on improved HSV and watershed algorithm. *IFAC-PapersOnLine*, *51*(17), 431-436.

[16] Miao, Z., Yu, X., Li, N., Zhang, Z., He, C., Li, Z., Sun, T. (2023). Efficient tomato harvesting robot based on image processing and deep learning. *Precision Agriculture*, *24*(1), 254-287.

[17] Seo, D., Cho, B. H., Kim, K. C. (2021). Development of monitoring robot system for tomato fruits in hydroponic greenhouses. *Agronomy*, *11*(11), 2211.

[18] Solimani, F., Cardellicchio, A., Dimauro, G., Petrozza, A., Summerer, S., Cellini, F., Renò, V. (2024). Optimizing tomato plant phenotyping detection: Boosting YOLOv8 architecture to tackle data complexity. *Computers and Electronics in Agriculture*, *218*, 108728.

[19] Tsai, F. T., Nguyen, V. T., Duong, T. P., Phan, Q. H., Lien, C. H. (2023). *Tomato Fruit Detection Using Modified Yolov5m Model with Convolutional Neural Networks. Plants 2023, 12, 3067*.

[20] Wang, S., Shao, Z., Zhang, Y. (2024). Enhanced precision in greenhouse tomato recognition and localization: A study leveraging advances in Yolov5 and binocular vision technologies. *Quality Assurance and Safety of Crops & Foods*, *16*(3), 67-81.

[21] Xu, D., Xiong, H., Liao, Y., Wang, H., Yuan, Z., Yin, H. (2024). EMA-YOLO: a novel target-detection algorithm for immature yellow peach based on YOLOv8. *Sensors*, *24*(12), 3783.

[22] Yang, Ren, Xujun, Chen, Lei, Wang. (2024), Research on weakly supervised directed target detection algorithm based on cross-space multi-scale (基于跨空间多尺度的弱监督有向目标检测算法研究), *Laser Journal*, 45(7), 63-70.
[23] Yin, X., Chen, L. (2024). Image object detection method based on improved faster R-CNN. *Journal of Circuits, Systems and Computers*, *33*(07), 2450130.
[24] Zeng, T., Li, S., Song, Q., Zhong, F., Wei, X. (2023). Lightweight tomato real-time detection method based on improved YOLO and mobile deployment. *Computers and electronics in agriculture*, *205*, 107625.
[25] Zhao, Z., Chen, S., Ge, Y., Yang, P., Wang, Y., Song, Y. (2024). RT-DETR-tomato: Tomato target detection algorithm based on improved RT-DETR for agricultural safety production. *Applied Sciences*, *14*(14), 6287.
[26] Zheng, T., Jiang, M., Li, Y., Feng, M. (2022). Research on tomato detection in natural environment based on RC-YOLOv4. *Computers and Electronics in Agriculture*, *198*, 107029.