YOLO11n-DRE: A METHOD FOR MATURITY DETECTION AND ACCURATE COUNTING OF SINGLE TRUSS TOMATO FRUITS IN COMPLEX UNSTRUCTURED ENVIRONMENTS

/

YOLO11n-DRE：复杂非结构化环境下串番茄单果成熟度检测与精准计数方法

Baofan CHEN, Yuhao HAO, Bingjun CHEN, Shuaishuai CUI, Yaqi YAN, Guozhu SONG
College of Software, Shanxi Agricultural University, Taigu, Shanxi / China;
Corresponding authors: Guozhu SONG; Tel: 03546287085; E-mail: songguozhu@sxau.edu.cn
DOI: https://doi.org/10.35633/inmateh-78-03

Keywords: *Maturity detection, YOLO11, Truss tomato, Fruit counting, Harvest assessment*

ABSTRACT

Accurate perception of the maturity of individual truss tomato fruits and in-situ counting in unstructured protected horticulture scenarios are critical prerequisites for driving selective automated harvesting, yield prediction and improving the level of refined management. Affected by factors such as the small fruit size and the complexity of the natural growth environment, vision-based maturity detection still faces considerable challenges. This paper proposes an improved method for maturity detection and counting of individual truss tomato fruits based on YOLO11n. Under the principle of maintaining a lightweight design, a fine-grained feature stream based on the P2 layer is developed, and the DySample operator is integrated to optimize the quality of feature fusion. Combined with the Selective Feature Refinement Module (EMA) and a four-head detector with full-scale coverage, the proposed method aims to maximize the model's representation capability in capturing small long-distance targets and in dense occluded environments. A dataset of individual truss tomato fruits with three maturity labels (immature, turning, ripe) is constructed, and systematic comparative experiments are conducted on the original YOLO11n and the improved model. The experimental results show that the improved YOLO11n-DRE outperforms the original YOLO11n model in maturity detection accuracy, with P, R and mAP@0.5 increased by 0.65%, 0.63% and 1.07% respectively, and the model parameters reduced by 8.5%. This method demonstrates excellent detection performance, and provides a reference model and technical prerequisite for the maturity detection and yield estimation of individual truss tomato fruits.

摘要

在非结构化设施园艺场景中，实现串番茄单果成熟度的精准感知与原位计数，是驱动选择性自动化采收、产量预测及提升精细化管理水平的关键前提。受果实尺度小以及自然生长环境复杂等因素影响，基于视觉的成熟度检测仍面临较大挑战。本文提出了一种基于 YOLO11n 改进的串番茄单果成熟度检测与计数方法。在维持轻量化设计原则下，开发基于 P2 层的细粒度特征流，集成 DySample 算子以优化特征融合质量。配合选择性特征精炼模块（EMA）与全尺度覆盖的四头检测器，旨在最大化模型在远距离小目标捕捉及密集遮挡环境下的表达能力。构建了包含三类成熟度标签（未熟、转色、成熟）的串番茄单果数据集，对原版 YOLO11n 与改进模型进行了系统对比实验。实验结果表明，改进后的 YOLO11n-DRE 在成熟度检测精度方面优于原始 YOLO11n 模型，在 P、R、mAP@0.5 等分别提升了 0.65%、0.63%、1.07%，模型参数减少了 8.5%。该方法展现出卓越的检测性能，为串番茄单果成熟度的检测和产量估算提供了参考模型和技术前提。

INTRODUCTION

Truss tomatoes are one of the important crops in China, with their planting area and output increasing steadily in recent years, which plays a vital role in ensuring farmers' income and residents' nutritional supply (*Yang et al., 2025*). Accurate acquisition of information on the fruit set number and maturity distribution of truss tomatoes is a key basis for guiding harvest scheduling, optimizing supply chain management and evaluating planting benefits. However, traditional manual yield estimation methods are not only labor-intensive and inefficient, but also prone to statistical errors due to subjective factors (*Zhu et al., 2026*), making it difficult to meet the demand for high-throughput and real-time phenotypic data in large-scale planting. Therefore, the construction of an automated detection and counting system based on machine vision has become a research hotspot in the current field of agricultural engineering (*Tang et al., 2026*).

In the intelligent picking process of truss tomatoes, the accurate identification of individual fruit maturity is a critical prerequisite for precise picking and rational decision-making (*Wang et al., 2024*).

The ripening of fruits on the same truss tomato cluster is obviously asynchronous. *(Gong et al., 2025)*. Improper picking timing, such as premature picking, will lead to an increase in the proportion of immature fruits and a decline in commodity value due to excessive cluster pruning; delayed picking, on the other hand, is likely to cause mechanical damage to fruits, reduced storage performance and increased risk of postharvest diseases, thereby significantly lowering the commodity rate and causing economic losses *(Cai et al., 2021).* Consequently, achieving accurate and stable identification of individual truss tomato fruit maturity in natural growth environments is of great significance for improving picking efficiency, ensuring fruit quality and enhancing production benefits.

In recent years, the boom in deep learning methods (DLMs) has provided a brand-new technical approach for crop phenotyping analysis. In particular, one-stage object detection algorithms represented by YOLO (You Only Look Once) have become the mainstream architecture in the field of agricultural computer vision due to their excellent balance between inference speed and accuracy. To address the occlusion challenge in complex field environments, the academic community has conducted extensive research. *Mbouembe et al. (2024)* recently proposed an improved YOLOv5s architecture, termed SBCS-YOLOv5s, which integrates attention mechanisms with lightweight convolutions. This approach maintains high detection accuracy while significantly reducing computational costs, offering an efficient vision solution for tomato-harvesting robots. *Zheng et al. (2024)* addressed the unstructured nature of greenhouse environments by proposing an improved YOLOv8-Tomato model. By integrating a dynamic BiFormer attention mechanism into the backbone network and optimizing the bounding box regression loss function (MPDIoU), the authors effectively mitigated the missed detection of fruits caused by high-density occlusion and varying illumination. This work demonstrates the robustness of the YOLOv8 architecture in complex greenhouse scenarios. Through comparative experiments on Jetson Orin NX and Raspberry Pi 5, *Rey et al. (2025)* revealed the non-linear relationship between model depth, input resolution, and Frame Per Second (FPS). They found that although YOLOv8s offers higher detection accuracy, the INT8-quantized YOLOv8n model often provides a better "accuracy-latency" balance in most real-time critical scenarios, offering significant guidance for designing low-latency edge intelligence systems. In addition, to solve the problem of feature blurring caused by uneven illumination, *Liu et al., (2023)*, designed a multi-scale illumination adaptive module, which effectively improved the stability of fruit recognition under dynamic greenhouse illumination conditions.

Although existing studies have made remarkable progress in single object recognition and model lightweighting, they still face the dual challenges of difficult individual fruit segmentation and inaccurate counting when targeting truss tomatoes as a specific object. Unlike single tomatoes, truss tomato fruits are distributed in high-density clusters with a high degree of physical adhesion and mutual occlusion between fruits. Current detectors tend to identify the entire fruit cluster, which is prone to missing and repeated detection in dense areas. *Gao et al., (2021),* proposed an innovative "Mutual Supervision" paradigm at ICCV 2021. This method breaks the traditional one-way supervision pattern by dividing anchors within the detection head into two subsets, utilizing their mutual prediction information during training to dynamically guide label assignment. This strategy not only avoids the tedious tuning of manual hyperparameters but also enables the detector to adaptively focus on high-quality prediction samples, significantly improving detection accuracy. In related research on berry crops, *Sozzi et al., (2023)*, used an improved YOLO model to detect grape clusters, but also noted that the model's ability to capture subtle color features still needs to be improved when distinguishing the maturity of individual berries within the cluster. Song et al. (2024) further fused an improved YOLOv8 algorithm with RGB-D depth information. They designed a specialized detection network for truss tomato picking points, embedding the SE (Squeeze-and-Excitation) module into the detection head to significantly enhance the extraction of fine stem features. This study not only achieved a 91.3% success rate in picking point recognition but also validated the great potential of fusing vision and depth information for spatial localization tasks involving truss tomatoes. *Deng et al., (2024),* compared the performance of YOLOv8 and YOLOv9 in blueberry canopy images and successfully realized the integrated processing of detection, counting and maturity evaluation of tiny fruits. In the field of truss tomato research, *Deng et al., (2024),* further expanded the detection dimension and proposed a deep learning framework based on key point detection, which can accurately locate picking points while identifying truss tomato fruits, providing fine visual feedback for automated operations.

Ma et al., (2024), systematically summarized the state-of-the-art in this field, highlighting that DL models, represented by Convolutional Neural Networks (CNNs), not only automatically extract high-dimensional features but also demonstrate superior performance over traditional methods in handling unstructured environmental issues such as illumination changes, occlusion, and fruit overlapping.

In view of this, this study proposes an improved method for maturity detection and counting of individual truss tomato fruits based on YOLO11n. As an iteration of the YOLO series, YOLO11 has carried out more in-depth optimizations in the design of the Feature Pyramid Network (FPN) and Path Aggregation Network (PAN), with stronger small target capture and feature fusion capabilities. This study aims to enhance the model's ability to distinguish features of densely adhered fruits within the cluster by introducing advanced attention mechanisms, solve the problem of inaccurate counting and segmentation of traditional detectors under severe occlusion, and provide strong technical support for the accurate yield prediction of truss tomatoes and the refined operation of agricultural robots.

MATERIALS AND METHODS

Dataset Construction

The data samples were collected at Nonggu Tomato Town in Taigu District, Jinzhong City, Shanxi Province. Given the close-range shooting mode and the high-density arrangement of individual fruits on truss tomato clusters, specific requirements are imposed on the resolution and focal length selection of the camera. The data samples were captured using the rear camera of a smartphone, which has a resolution of 2778×1284 with 12 million pixels, and its telephoto camera features a 77 mm focal length—enabling high-quality, distortion-free image data collection from multiple positions and angles.

(a)Sunny (b)Cloudy (c)Overcast (d)Occluded (e)Dense (f)Complex

Fig. 1 – Sample images of truss tomatoes collected under different environmental backgrounds

In this study, a dataset for the maturity of individual truss tomato fruits was constructed to train a model capable of simultaneously detecting fruit maturity and counting the fruits. The data were collected from January 2024 to October 2025, during which most truss tomatoes in the greenhouse were at the half-ripe to ripe stage, making them suitable for harvest decision-making and yield estimation. Under diverse natural lighting conditions and protected agricultural environments, canopy and fruit cluster images of truss tomatoes were captured at different spatial scales; such data are of great value for constructing a highly robust model for tomato detection and maturity assessment. Image collection covered a variety of perspectives and scenarios, ranging from detailed close-ups of individual fruits and complete fruit clusters to the canopies of entire plants, so as to ensure the diversity of the dataset. Figure 1 shows sample images of truss tomatoes collected under different environmental backgrounds.

The content of this dataset is presented in Table 1, which contains a total of 2,785 images partitioned into training, validation and test sets at a ratio of 7:2:1. A cumulative total of 62,673 tomato instances were annotated, with detailed statistics compiled by maturity category.

Table 1

Annotation Quantity Statistics

Dataset Partition	Number of Images	orange	red	green	Total Number of Labels
Training	1,948	6,619	34729	12761	44,109
Validation	558	1,743	6861	3789	12,393
Test	279	811	3537	1823	6,171
Total	2,785	9173	35127	18373	62,673

All collected images were processed in accordance with standard operating procedures, and the open-source Python library LabelImg was used to complete Bounding Box annotation. To achieve refined evaluation of tomato maturity and reduce ambiguities in model learning, individual tomato instances were subdivided into three categories based on pericarp color features: Green (immature), Orange (turning stage), and Red (ripe), as shown in Figure 2. The annotation format is expressed as Equation (1):

$$L = (c, x, y, w, h) \quad (1)$$

In the equation, L denotes the annotated ground truth of a single tomato fruit; $c \in \{0,1,2\}$ represents the fruit maturity category (corresponding to Orange, Red and Green respectively); (x,y) are the normalized coordinates of the center point of the target bounding box; and w and h are the normalized width and height of the bounding box, respectively.

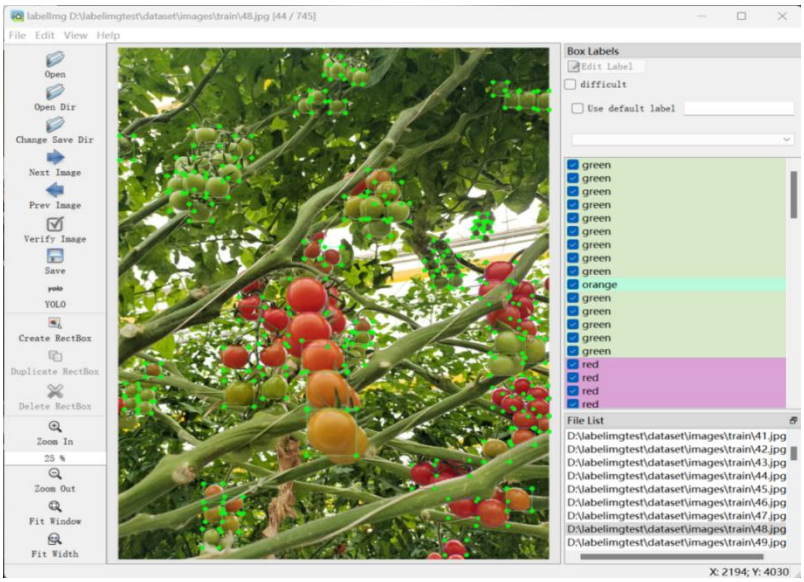

Fig. 2 – Annotation Demo Figure

YOLO11n-DRE Model

To enhance the object detection performance of lightweight detection models in complex scenarios, especially their ability to recognize and locate small-sized targets, this paper proposes a systematic enhanced improvement scheme for the YOLO architecture. Starting from five key stages—feature extraction, multi-scale expansion, feature fusion, feature refinement, and detection output—the scheme designs five mutually collaborative improvement modules and constructs a complete enhanced framework for small target detection.

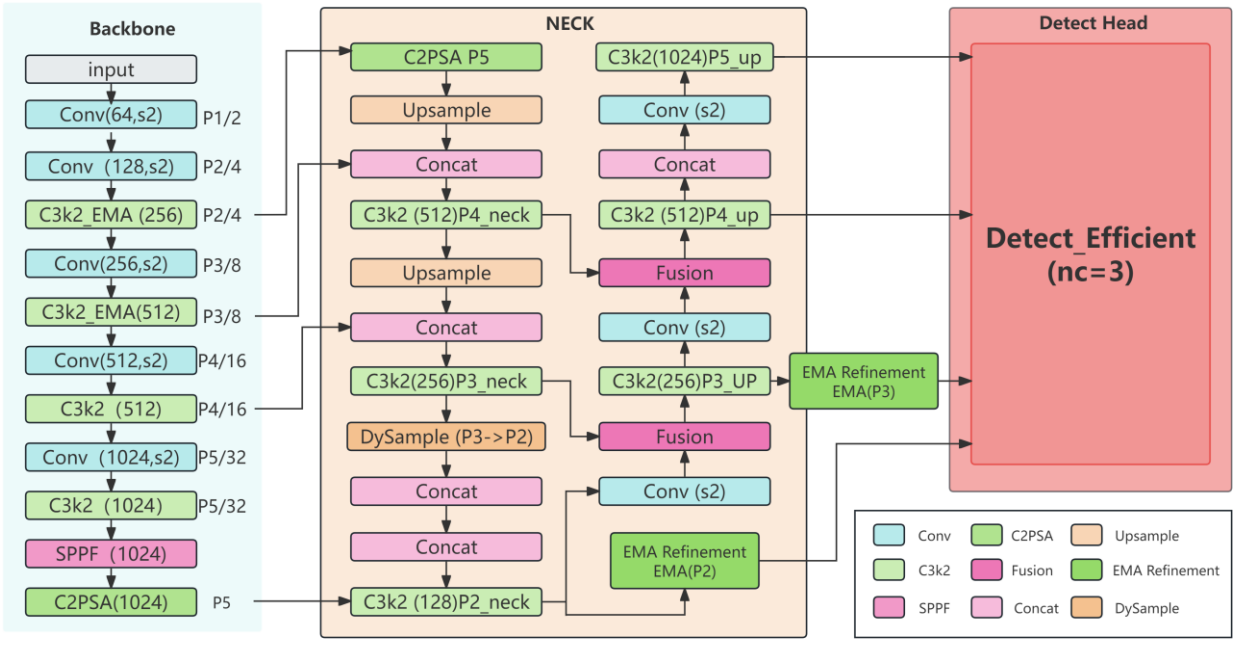

Fig. 3 – Flow Chart of YOLO11n-DRE

As shown in Figure 3, the model structure follows the design paradigm of a typical one-stage detector and consists of three parts: the Backbone, the Neck, and the Detect Head. The Backbone is responsible for multi-level feature extraction and incorporates an efficient multi-scale attention mechanism at the shallow stage to suppress background noise; the Neck adopts a top-down feature pyramid structure and extends to the high-resolution P2 layer to enhance the perception of tiny targets; finally, a high-efficiency four-scale Detect Head is used to complete multi-scale prediction. While maintaining the lightweight nature of the model, this framework significantly improves the model's robustness to tiny targets, dense targets and complex backgrounds, achieving a balance between accuracy and efficiency.

Backbone Enhancement Module (with EMA)

Lightweight detection models are often plagued by the problem of feature aliasing in complex environments, where high-frequency background noise and target boundaries are invariably encoded without discrimination. This phenomenon introduces a large amount of redundant information in the subsequent feature fusion stage, which in turn impairs localization accuracy and classification confidence—an issue that is particularly fatal for small target detection. To address this, as shown in Figure 4, the standard C3k2 module is reconstructed into the C3k2_EMA module integrated with the Efficient Multi-Scale Attention (EMA) mechanism *(Ouyang et al., 2023*) at the shallow and mid-shallow stages of the backbone network (corresponding to the P2/4 and P3/8 feature extraction layers). With the premise of keeping the feature resolution (i.e., stride) unchanged, this design endows the network with the ability to suppress noise and focus on salient regions at the early stage of feature extraction through adaptive recalibration of the responses in channel and spatial dimensions. By enhancing the selectivity for structural details such as edges and corners while filtering out irrelevant texture responses *(Hou et al., 2023)*, this improvement provides purer and more discriminative base features for multi-scale fusion in the Neck, thus effectively facilitating the stable convergence of the network and significantly improving the separability and localizability of tiny targets.

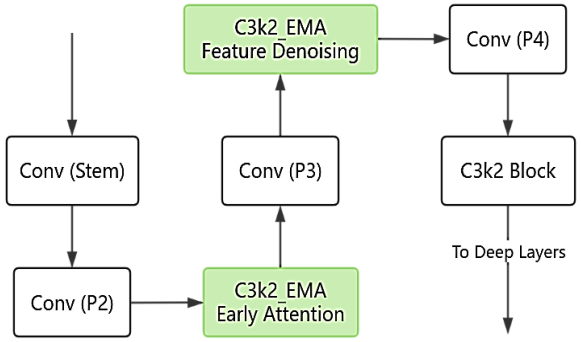

Fig. 4 – Structure Diagram of Backbone Enhancement with EMA

Micro-Target Perception Layer Extension Module (P2 Micro-Scale Layer Extension)

The original YOLO11 architecture performs predictions only at three scales (P3, P4, and P5, corresponding to strides of 8, 16, and 32). The spatial resolution of its finest-scale feature map remains insufficient for covering ultra-tiny targets (with a low pixel proportion), which creates an inherent recall bottleneck and constitutes a structural limitation. To address this issue, as shown in Figure 5, this module extends the topological structure of the top-down path in the Neck and further extends it to P2 (with a stride of 4), thus forming a complete high-resolution semantic transfer chain of P5→P4→P3→P2. By introducing the P2 branch, the model is able to reconstruct the geometric details of tiny targets at a denser grid scale, effectively alleviating the problem of spatial information dissipation caused by multiple downsampling operations *(Zhu et al., 2021*). This extension represents a fundamental structural enhancement: it significantly improves the model's pixel-level modeling capability for small-sized targets (low-pixel targets), dense targets, and fine-grained targets, effectively increases the recall rate of small targets, and minimizes the loss of spatial detail information caused by multiple downsampling operations.

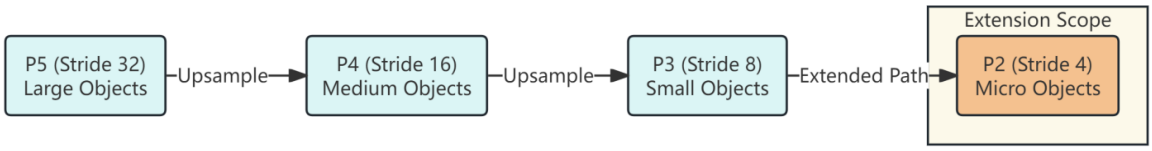

Fig. 5 – Structure Diagram of P2 Micro-Scale Layer Extension

Content-Aware Dynamic Upsampling Module (Content-Aware DySample)

In the top-down feature fusion path, especially during the generation of the newly added P2 branch, the use of traditional nearest-neighbor upsampling, despite its low computational overhead, results in limited capability for recovering detailed structures and is prone to producing jagged artifacts and semantic information voids. This may lead to insufficient information quality of features in the P2 branch even though it has a higher spatial resolution, ultimately impairing the boundary localization accuracy of small targets. As shown in Figure 6, in the critical upsampling step for generating P2 features (i.e., P3→P2), this module replaces the conventional nearest-neighbor upsampling with a content-aware dynamic upsampling operator (DySample). This operator can adaptively perform feature reorganization and interpolation based on the local content structure of input features, thereby improving the recovery fidelity of detailed features. This operator-level improvement aims to enhance the effective information density of the high-resolution branch, making the generated P2 feature map not only larger in size but also clearer in content. In particular, it can preserve sharper edges and more distinct local contrast relationships, which in turn directly improves the localization accuracy (e.g., IoU metric) and classification confidence of small targets.

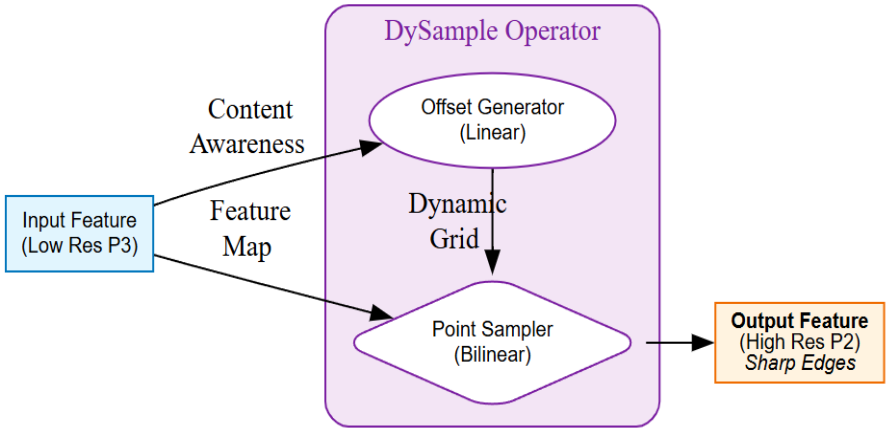

Fig. 6 – Structure Diagram of Content-Aware DySample

Pre-Head Feature Refinement Module (Pre-Head Feature Refinement)

Features after multi-scale fusion, especially shallow features associated with small targets such as P2 and P3, may still be mixed with redundant background noise and conflicting information from different scales. These features contain both fine-grained details of targets and easily disruptive texture noise; if directly fed into the Detect Head without processing, the noise will be amplified by the classification and regression branches, leading to false detections or a drop in confidence. To address this issue, as shown in Figure 7, drawing on the advanced concept of enhancing information interaction between the Neck and the Detect Head proposed in Gold-YOLO (*Wang et al., 2023*), this module presents a lightweight Pre-Head Refinement strategy, which inserts a lightweight, plug-and-play EMA refinement module into the two key branches of P2 and P3.

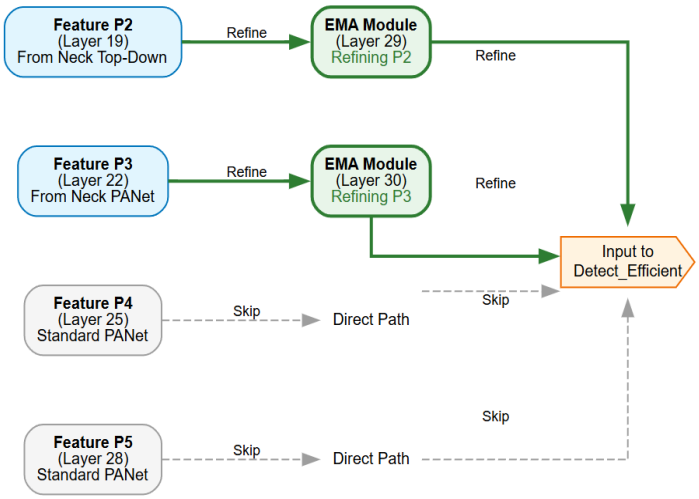

Fig. 7 – Structure Diagram of Pre-Head Feature Refinement

Without altering the main structure of the network backbone and prediction heads, this module performs secondary filtering and channel/spatial weighting on the features to be fed into the Detect Head, thereby focusing on and enhancing responses related to small targets and suppressing irrelevant interference. This pre-detection purification strategy makes the input features received by the prediction heads more discriminative, helping to reduce false positive detections and improve the detection confidence of hard small targets. Meanwhile, since the refinement module only acts on the P2 and P3 branches, the additional computational overhead it introduces is strictly controlled, achieving a cost-effective balance between performance gains and computational costs.

Efficient 4-Scale Detection Head Module (Efficient 4-Scale Detection Head)

After the introduction of the P2 prediction scale, the detection head is required to process four-scale inputs from P2, P3, P4 and P5. Directly adopting the original three-scale detection head design would lead to a significant increase in the number of parameters and computational load, compromising the model's inference speed and real-time performance. Meanwhile, feature redundancy among multi-scale predictions may also increase the difficulty of model optimization.

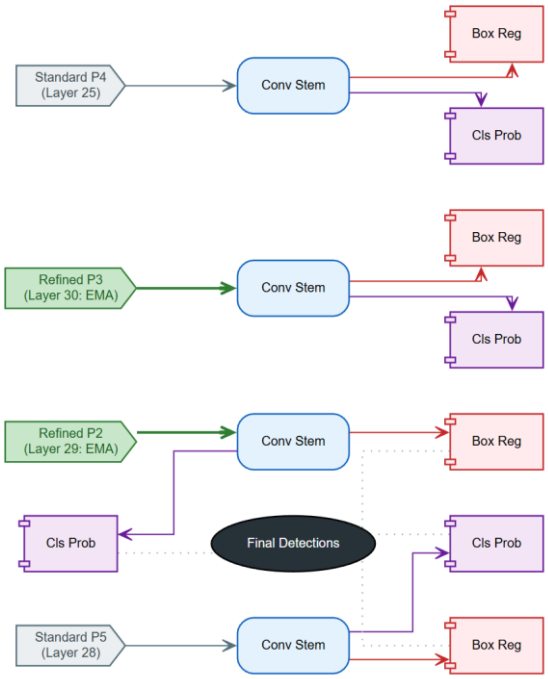

Fig. 8 – Structure Diagram of Efficient 4-Scale Detection Head

As shown in Figure 8, inspired by RTMDet (C et al., 2023), an advanced real-time detector architecture, this module constructs an efficient four-scale detection head (Detect_Efficient) to replace the original detection head and adapt it to the four-scale inputs from P2 to P5. Structurally, this efficient detection head is optimized for multi-scale prediction tasks—for instance, it adopts a more lightweight prediction branch design and a more efficient approach to parameter sharing and organization—aiming to achieve effective prediction for four scales at the minimum additional cost. The intended benefit of this module is that it can effectively control the overall computational complexity and parameter count of the model while achieving a significant improvement in small target detection performance brought by the P2 extension. Thus, the improved model attains a better balance among detection accuracy, inference speed and model size, making it suitable for application scenarios with stringent requirements for both real-time performance and accuracy, such as UAV aerial photography, autonomous driving and mobile vision.

RESULTS

Experimental Setup

The configuration parameters of the experimental equipment used in this experiment are as follows: the processor is 25 vCPU Intel(R) Xeon(R) Platinum 8470Q; the graphics card is NVIDIA RTX 5090 (32 GB); the graphics card driver version is NVIDIA-SMI 580.76.05; the memory is DDR5 90 GB; the operating system is Ubuntu 22.04.3 LTS; the depth camera is Intel RealSense D455i; the development language is Python 3.11.14;

the CUDA version of the configured environment is CUDA 12.8; the Anaconda version is 24.1.2. To verify the effectiveness of the improved YOLO11n-DRE model in the actual detection scenario of truss tomatoes, images actually collected in greenhouses that were not involved in model training were employed as inputs to simulate practical conditions, and conducted a comparative experiment with the base model YOLO11n. As can be seen from the comparison of detection results in Figure 9, for the same frame of image, the YOLO11n-DRE model achieved a slightly higher recognition confidence than YOLO11n when the truss tomatoes were in a clear growth state; in the case of stem occlusion, the YOLO11n-DRE model exhibited better detection performance and was able to detect more tomatoes. Meanwhile, the model reached a frame rate (FPS) of 370, which meets the real-time detection requirements for truss tomatoes.

In Figure 9, the left column presents the results of YOLO11n, while the right column shows those of YOLO11n-DRE.

Fig. 9 – Comparison of Detection Results

Performance Evaluation Metrics

The complexity of detection models is evaluated using Parameters (Params) and Giga Floating-Point Operations Per Second (GFLOPs) as evaluation metrics. The detection speed of a model is measured by Frames Per Second (FPS). Mean Average Precision (mAP) is used to represent the average accuracy, which is calculated from the precision and recall of the prediction model; the average precision in this study can be denoted by the AP value, which refers to the area enclosed by the Precision-Recall (P-R) curve. In addition, the calculation formulas for Precision (P), Recall (R) and F1-score are shown in Equations (2) to (4) below.

$$Precision\ (P) = \frac{TP}{TP+FP} \tag{2}$$

$$Recall\ (R) = \frac{TP}{TP+FN} \tag{3}$$

$$Flscore = 2 \times \frac{Precision \times Recall}{Precision+Recall} \tag{4}$$

In this study, the target objects to be detected are defined as the positive class and all other categories as the negative class, with the corresponding evaluation metrics defined as follows: a True Positive (TP) represents a positive instance correctly predicted as positive, a False Negative (FN) a positive instance incorrectly predicted as negative, a False Positive (FP) a negative instance incorrectly predicted as positive, and a True Negative (TN) a negative instance correctly predicted as negative. To evaluate inference efficiency, the model was executed over 500 iterations on the test samples, and the detection speed was measured in frames per second (FPS) based on the average runtime.

Comparison of Experimental Results of Mainstream Models

To comprehensively evaluate the performance of the YOLO11n-DRE model proposed in this paper, comparative experiments were conducted between this model and a variety of mainstream and cutting-edge object detection models, including Faster R-CNN, RT-DETR (with the RT-DETR-L model adopted), the YOLOv5/v8/v9/v10 series, and YOLO11n. The comparison results are presented in Table 2. Simultaneously, the mAP@0.5 and mAP@0.5:0.95 results for different models are presented in Figure 10. The experimental results strongly demonstrate that the improved YOLO11n-DRE algorithm achieves outstanding performance and features prominent advantages of fast recognition speed and high accuracy.

Table 2

Experimental results of different models

Model	P	R	mAP50	mAP50-95	F1-score	GFlops(G)	Param
	[%]	[%]	[%]	[%]	[%]		
Faster R-CNN	73.4	90.9	84.9	57.7	81.2	147.92	41.31
RT-DETR	76.4	72.4	75.8	52.3	74.3	100.6	28.45
YOLOv5n	77.5	76.8	81.9	55.8	77.1	4.1	2.6
YOLOv8n	77.8	78.6	83.6	58.1	78.2	6.8	3.2
YOLOv9t	76.4	79.3	83.2	58	77.8	6.4	2.0
YOLOv10n	77	78.1	83.3	58.2	77.5	6.5	2.3
YOLO11n	77.5	79.7	83.8	59.4	78.6	6.5	2.6
YOLO11n-DRE	78	80.2	84.7	59.2	79.2	6.8	2.38

In terms of detection accuracy

YOLO11n-DRE demonstrates significant advantages across multiple core metrics. As shown in Table 5, its mAP@0.5 reaches 0.847, ranking first among all compared one-stage detectors—an increase of 1.4 percentage points over YOLOv10n and 1.1 percentage points over YOLOv8n. This verifies the effectiveness of the DRE module in enhancing feature extraction and fusion. For mAP@0.5:0.95, a comprehensive metric for detection performance under high IoU thresholds, YOLO11n-DRE achieves a leading score of 0.592, 1.0 percentage points higher than YOLOv10n, which indicates the model's excellent stability in identifying tomato targets with varying degrees of overlap. In addition, the model strikes a favorable balance between Precision (P=0.78) and Recall (R=0.802), achieving the highest F1-score of 0.792 (1.0 percentage points higher than YOLOv8n). This manifests the model's remarkable efficacy in suppressing false detections and reducing missed detections, enabling comprehensive and accurate detection of target tomatoes.

In terms of model efficiency and the accuracy-computation trade-off

YOLO11n-DRE delivers an exceptional cost-performance ratio. Compared with Faster R-CNN, a classic two-stage detector, YOLO11n-DRE has a slightly lower mAP@0.5 (a decrease of 0.2 percentage points), but its computational load is merely 6.8 GFLOPs (approximately 4.6% of Faster R-CNN's 147.92 GFLOPs) and its parameter count is only 2.38M (around 5.8% of Faster R-CNN's 41.31M), which highlights the high efficiency inherent to one-stage detectors. When compared with lightweight models with similar computational loads: YOLO11n-DRE matches YOLOv8n in computational load (6.8 GFLOPs for both), yet it achieves a 1.1 percentage point increase in both mAP@0.5 and mAP@0.5:0.95 while reducing the parameter count by 25.6%. In comparison with YOLOv10n, although YOLO11n-DRE has a roughly 4.6% higher computational load, it gains a 1.4 percentage point improvement in mAP@0.5 and a 1.0 percentage point rise in mAP@0.5:0.95, proving that the additional computational resources are efficiently converted into enhanced detection accuracy.

A noteworthy in-depth analysis

Despite a slight fluctuation of 0.02 in mAP@0.5:0.95 compared with YOLO11n, this actually reflects the inherent geometric characteristics of tiny target detection tasks.

Since the model is specifically designed to enhance the capture capability for ultra-tiny targets (at the P2 level), which feature extremely low pixel resolution, a minor prediction deviation of just 1–2 pixels in the bounding box can lead to a significant drop in IoU@0.95. However, in practical agricultural applications such as tomato yield statistics and picking localization, the detection rate takes precedence over pixel-level bounding box regression accuracy. Experiments show that while maintaining a high mAP@0.5, the model preserves real-time inference speed through lightweight design, achieving a better balance between detection accuracy and engineering practicability for real-world applications.

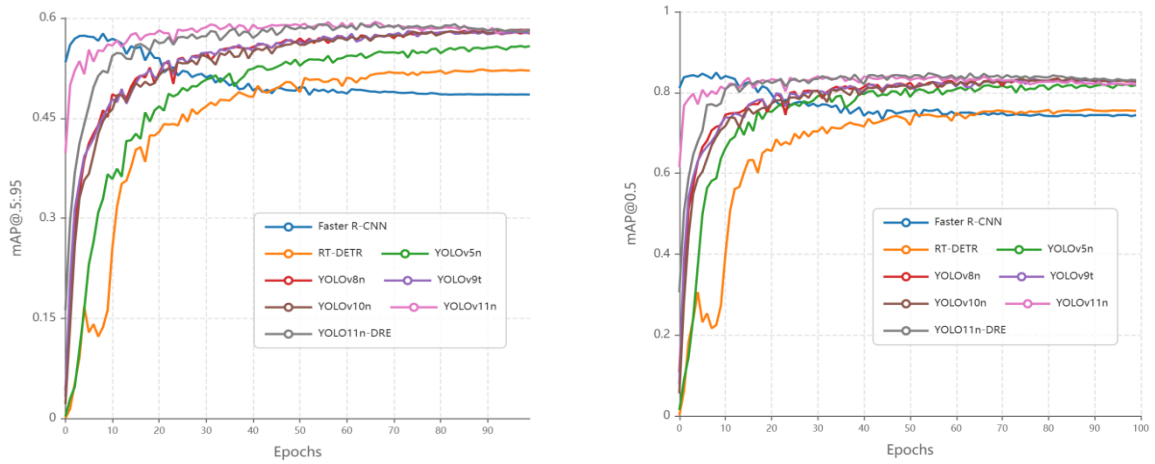

Fig. 10 – Comparison Chart of mAP@0.5 and mAP@0.5:0.95 Results

As shown in Fig. 11, a comparison of the two sets of Precision–Recall (PR) curves indicates that YOLO11n-DRE outperforms the baseline YOLO11n in both overall detection performance and category consistency. The all-classes curve is closer to the top-right corner across most recall intervals and exhibits a larger area under the curve, with mAP@0.5 increasing from 0.847 to 0.855, indicating improved comprehensive detection capability. At the category level, the AP values for the orange, red and green categories all show improvements, with a notable gain (+0.019) for the relatively challenging orange category, which demonstrates the model's stronger discriminative ability and robustness for complex samples.

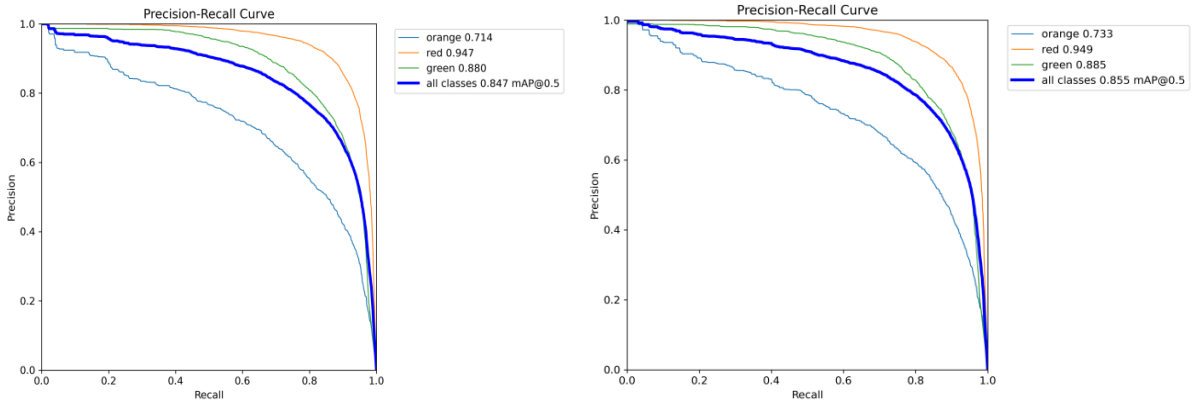

Fig. 11 – PR Comparison Chart of YOLO11n and YOLO11n-DRE

As shown in Figure 12, the comparison of the normalized confusion matrices reveals that YOLO11n-DRE outperforms the baseline model overall in category discrimination and missed detection suppression. The diagonal proportions for all three target categories are higher, indicating that the model has a more adequate characterization of target features and stronger intra-class consistency. Meanwhile, the proportion of misclassifying real targets as the background is significantly reduced, demonstrating the model's better detection capability and robustness for samples with weak textures and other challenging characteristics.

Moreover, the separability of some easily confused categories is improved. In summary, by increasing the accuracy rate, reducing missed detections and inter-class confusion, YOLO11n-DRE achieves more stable and reliable recognition performance in multi-category scenarios and is superior to the baseline network.

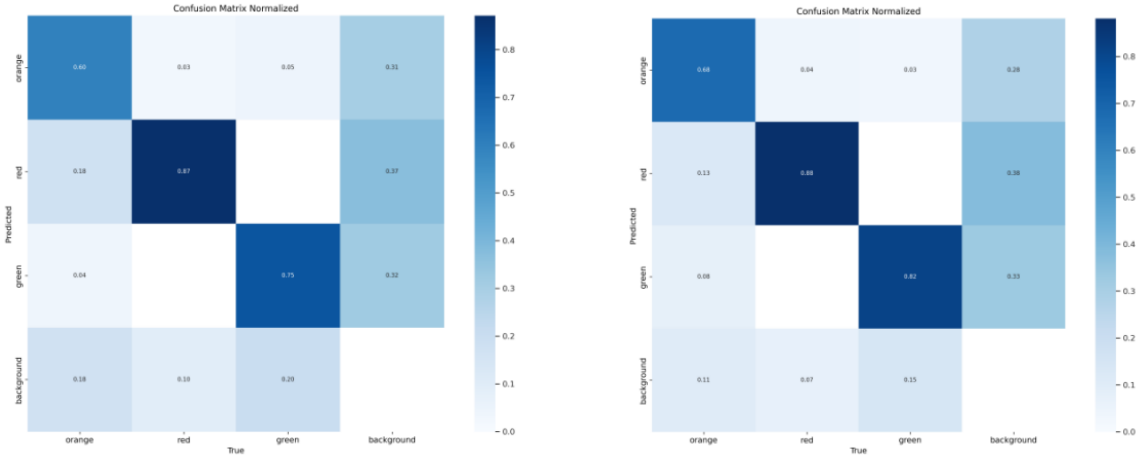

Fig. 12 – Confusion Matrix Comparison Chart of YOLO11n and YOLO11n-DRE

Counting Method Based on Detection Results

To address the urgent demand for accurate yield monitoring and agronomic decision-making of truss tomatoes in the protected horticulture environment, this study constructs a digital yield analysis system integrating multi-dimensional information perception, logic-gated counting and augmented reality visualization on the basis of the high-precision detection of YOLO11n-DRE. This system is designed to solve the challenges of low-confidence noise interference and underutilization of unstructured data in traditional visual counting (*Glukhikh et al., 2025*), and realize the automatic parsing of phenotypic parameters by defining rigorous post-processing logic.

Specifically, the system adopts a three-stage cascaded processing mechanism of detection-filtering-quantification. First, it parses the tensor information output by the model and extracts the category label c, confidence score s and bounding box coordinates B of all candidate targets. Second, a Confidence Gating Mechanism is introduced, which eliminates redundant background noise and low-quality predictions by setting an adaptive threshold τ (set to 0.5 in this study) and performs category-level accumulation of valid targets. For the fruit count N_k of the k-th ripeness class (e.g., ripe, color turning, unripe), its calculation model is defined as Equation (5):

$$N_k = \sum_{i=1}^{M} \mathbb{I}\,(c_i = k) \cdot \mathbb{I}(s_i \geq \tau) \tag{5}$$

where M is the total number of original detection boxes, and $I(\cdot)$ is the Indicator Function, which takes the value of 1 if the condition is satisfied and 0 otherwise. On this basis, to quantify the distribution characteristics of fruits at different ripeness levels in the canopy, a ripeness distribution ratio model R_k (Equation 6) is further constructed to provide real-time feedback on the current harvest potential (*Dubey et al., 2025*):

$$R_k = \frac{N_k}{\sum_{j\in\{mature,breaking,green\}} N_j} \times 100\% \tag{6}$$

At the interaction level, the system innovatively designs a dual-layer visual mapping architecture. The bottom layer adopts instance-level semantic enhancement, using the red, orange and green spectral colors to map ripe, color-turning and unripe fruits respectively. It fuses the "category-confidence" joint label on top of the bounding box via a translucent mask, enabling intuitive information indexing while ensuring that target features remain unobscured.

The top layer embeds a head-up display (HUD) as a floating statistical dashboard, which refreshes the count N_k and proportion R_k of each category in real time and dynamically in a tabular format. The system supports batch pipeline operations and features an end-to-end processing capability that converts image sequences into structured yield data. Validated on the test set, this logic can effectively suppress false detections and achieve highly robust counting accuracy without the need for additional parameter fine-tuning, providing reliable data support for the operation scheduling of tomato harvesting robots and the formulation of graded sales strategies. The operational effect of the system is shown in Figure 13.

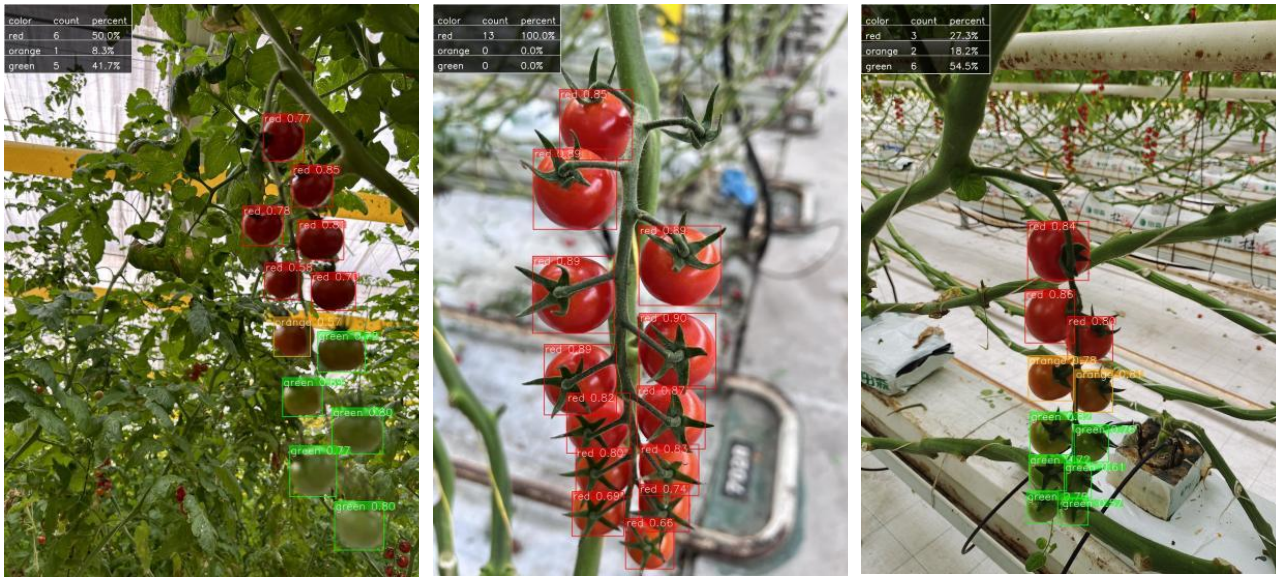

Fig. 13 – Implementation Effect Diagram of the Counting Script

CONCLUSIONS

Aiming at the maturity detection and counting tasks of dense single truss tomatoes in the complex greenhouse environment, and addressing the challenges such as small targets, high density, occlusion and background interference, this paper proposes YOLO11n-DRE based on the YOLO11n baseline, and constructs a dataset covering three maturity levels (green/orange/red) for model training and testing. While retaining a lightweight backbone, the proposed method enhances the small target representation and anti-interference capability in dense regions through the designs of fine-grained feature preservation, cross-scale aligned fusion and attention reweighting. Comparative experiments show that, compared with YOLO11n, this method achieves an improvement in mAP@0.5, with consistent upward trends also observed in Precision and Recall; the computational load increases slightly from 6.5 GFLOPs to 6.8 GFLOPs, reflecting an acceptable accuracy-overhead trade-off.

To directly convert detection outputs into usable statistical information, this paper further implements a lightweight counting and visualization system: it performs confidence threshold filtering and category cumulative statistics on the detection results, and outputs the quantity and proportion of tomatoes at each maturity level. At the instance level, color-coded bounding boxes and confidence labels are adopted to intuitively display the recognition results; at the image level, a translucent statistical table embedded in the upper left corner is used to present the "category-quantity-percentage" in real time. This workflow supports batch processing and real-time output, and the counting results in the test show high consistency with manual annotations, which can provide an intuitive basis for greenhouse patrol inspection, yield estimation and harvesting decision-making.

This study still has several limitations: the dataset coverage is still limited in terms of variety differences, greenhouse structures and lighting conditions. In the follow-up, it is necessary to expand the sample size and scene diversity, and introduce data augmentation strategies that are more in line with the greenhouse imaging mechanism to improve the cross-domain generalization capability (Gong et al., 2025). In extreme scenarios such as highly adhered targets, severe occlusion and ambiguous boundaries, missed detections or localization deviations may still occur, which in turn have a cumulative impact on the counting results; further exploration can be conducted on robust counting strategies and occlusion recovery methods for dense targets. The current evaluation system is still dominated by detection metrics. In the follow-up, a more targeted comprehensive evaluation can be constructed from the "detection-counting-yield estimation/harvesting decision-making" chain, and deployment and time-delay tests on different computing power platforms can be carried out to verify the engineering implementation effect.

ACKNOWLEDGEMENT

This study was supported by the following projects: the Key R&D Program of Shanxi Provincial Department of Science and Technology (No. 202102140601015), and the 2023 Science and Technology Innovation Project of Jinzhong National Agricultural High-tech Industrial Development Zone (Taigu National Science and Technology Innovation Center).

REFERENCES

[1] Cai Y., Takeda F., Foote B., DeVetter L.W. (2021). Effects of machine-harvest interval on fruit quality of fresh market northern highbush blueberry, *Horticulturae*, vol.7, no.8, 245, Basel/Switzerland. DOI: https://doi.org/10.3390/horticulturae7080245.

[2] Deng B., Lu Y., Li Z. (2024). Detection, counting, and maturity assessment of blueberries in canopy images using YOLOv8 and YOLOv9, *Smart Agricultural Technology*, vol.9, 100620, Amsterdam/Netherlands.

[3] Deng L., Ma R., Chen B.F. (2025). A detection method for synchronous recognition of string tomatoes and picking points based on keypoint detection, *Frontiers in Plant Science*, vol.16, 1614881, Lausanne/Switzerland. DOI: https://doi.org/10.3389/fpls.2025.1614881.

[4] Dubey P., Waghodekar P., Lahane S.P., Bhagat D., Dubey P., Zakariah M. (2025). TomatoRipen-MMT: transformer-based RGB and NIR spectral fusion for tomato maturity grading, *Scientific Reports*, vol.16, no.1, pp.2714-2714, London/United Kingdom. DOI: https://doi.org/10.1038/S41598-025-32522-9.

[5] Gao Z., Wang L., Wu G. (2021). Mutual supervision for dense object detection, *Proceedings of the 2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp.3621-3630, Montreal/Canada. DOI: https://doi.org/10.1109/ICCV48922.2021.00362.

[6] Glukhikh I., Glukhikh D., Gubina A. (2025). Deep learning method with domain-task adaptation and client-specific fine-tuning YOLO11 model for counting greenhouse tomatoes, *Applied System Innovation*, vol.8, no.3, pp.71-71, Basel/Switzerland. DOI: https://doi.org/10.3390/ASI8030071.

[7] Gong R.J., Cheng L.J., Zhang Y.B., Feng Z.X. (2025). Research on a lightweight tomato ripeness detection method based on SFH-YOLOv11, *INMATEH - Agricultural Engineering*, vol.77, no.3, pp.1482-1493, Bucharest/Romania. DOI: https://doi.org/10.35633/inmateh-77-118.

[8] Hou Q., Zhou D., Feng J. (2021). Coordinate attention for efficient mobile network design, *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp.13713-13722, Nashville/USA. DOI: https://doi.org/10.1109/CVPR46437.2021.01350.

[9] Liu G., Nouaze J.C., Mbouembe P.L., Kim J.H. (2023). YOLO-Tomato: a robust algorithm for tomato detection based on YOLOv3 and adaptive illumination, *Expert Systems with Applications*, vol.215, 119401, Amsterdam/Netherlands. DOI: https://doi.org/10.1016/j.eswa.2022.119401.

[10] Lyu C., Zhang W., Huang H., Zhou Y., Wang Y., Liu Y., Zhang S., Chen K. (2023). RTMDet: an empirical study of designing real-time object detectors, *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp.4381-4390, Vancouver/Canada.

[11] Ma J., Li M., Fan W., Liu J. (2024). State-of-the-art techniques for fruit maturity detection, *Agronomy*, vol.14, 2783, Basel/Switzerland. DOI: https://doi.org/10.3390/agronomy14122783.

[12] Ouyang D., He S., Zhang G., Luo M., Guo H., Zhan J., Huang Z. (2023). Efficient multi-scale attention module with cross-spatial learning, *Proceedings of the 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp.1-5, Rhodes Island/Greece. DOI: https://doi.org/10.1109/ICASSP49357.2023.10096516.

[13] Rey L., Bernardos A.M., Dobrzycki A.D., Carramiñana D., Bergesio L., Besada J.A., Casar J.R. (2025). A performance analysis of You Only Look Once models for deployment on constrained computational edge devices in drone applications, *Electronics*, vol.14, no.3, 638, Basel/Switzerland. DOI: https://doi.org/10.3390/electronics14030638.

[14] Song G., Wang J., Ma R., Shi Y., Wang Y. (2024). Study on the fusion of improved YOLOv8 and depth camera for bunch tomato stem picking point recognition and localization, *Frontiers in Plant Science,* vol.15, 1447855, Lausanne/Switzerland. DOI: https://doi.org/10.3389/fpls.2024.1447855.

[15] Sozzi M., Cantalamessa S., Cogato A., Kayad A., Marinello F. (2022). Automatic bunch detection in white grape varieties using YOLOv3, YOLOv4, and YOLOv5 deep learning algorithms, *Agronomy*, vol.12, 319, Basel/Switzerland. DOI: https://doi.org/10.3390/agronomy12020319.

[16] Tang Z.Y., He Y.C., Tang Z.H. (2025). Research status and development trends of orchard harvesting machinery (果园采摘机械的研究现状与发展趋势). *Acta Agriculturae Zhejiangensis*, pp.1-13, Hangzhou/China.

[17] Touko Mbouembe P.L., Liu G., Park S., Kim J.H. (2024). Accurate and fast detection of tomatoes based on improved YOLOv5s in natural environments, *Frontiers in Plant Science*, vol.14, 1292766, Lausanne/Switzerland. DOI: https://doi.org/10.3389/fpls.2023.1292766.

[18] Wang A., Qian W., Li A. (2024). NVW-YOLOv8s: an improved YOLOv8s network for real-time detection and segmentation of tomato fruits at different ripeness stages, *Computers and Electronics in Agriculture*, vol.219,108833, Amsterdam/Netherlands.

[19] Wang C., He W., Nie Y., Guo J., Liu C., Wang Y., Han K. (2023). Gold-YOLO: efficient object detector with gather-and-distribute mechanism, *Advances in Neural Information Processing Systems (NeurIPS)*, vol.36, pp.56250-56266, New Orleans/USA.

[20] Yang J. (2025). A brief discussion on green facility cultivation of tomatoes in modern agriculture (浅谈番茄现代农业绿色设施种植). *Henan Agriculture,* no.20, pp.16-18, Zhengzhou/China.

[21] Zheng S., Jia X., He M., Zheng Z., Lin T., Weng W. (2024). Tomato recognition method based on the YOLOv8-Tomato model in complex greenhouse environments, *Agronomy*, vol.14, no.8, 1764, Basel/Switzerland. DOI: https://doi.org/10.3390/agronomy14081764.

[22] Zhu A.Q., Tian W.J., Li M.F. (2025). Review and prospect of the application of artificial intelligence technology in agricultural production in China—taking its application in tomato production as an example (我国人工智能技术在农业生产领域中的应用研究综述及展望——以番茄生产中的应用为例). *Journal of Northeast Agricultural Sciences,* pp.1-9, Changchun/China.

[23] Zhu X., Lyu S., Wang X., Zhao Q. (2021). TPH-YOLOv5: improved YOLOv5 based on Transformer Prediction Head for object detection on drone-captured scenarios, *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) Workshops*, pp.2778-2788, Montreal/Canada. DOI: https://doi.org/10.1109/ICCVW54120.2021.00312.