applied sciences

MDPI

Article

MC-RT-DETR: A Lightweight Real-Time Cotton Top Bud Detection Model Based on Improved RT-DETR

Yibulayin Kuwan [1,3], Gulinazi Ailimujiang [2,3,4,*], Xiaolong Qi [2,3,4], Xin Liu [1,3], Wangwang Cao [2,3] and Mayire Hudabaierdi [1,3]

1 School of Electronic Engineering, Yili Normal University, Yining 835000, China; 20250050207@ylnu.edu.cn (Y.K.); 20240050189@ylnu.edu.cn (X.L.); 20250050208@ylnu.edu.cn (M.H.)
2 School of Network Security and Information Technology, Yili Normal University, Yining 835000, China; qixl@ylnu.edu.cn (X.Q.); 20250170680@ylnu.edu.cn (W.C.)
3 State Key Laboratory for Novel Software Technology, Nanjing University, Nanjing 210093, China
4 Yili Key Laboratory of Intelligent Computing Research and Application, Yining 835000, China
* Correspondence: glnz@ylnu.edu.cn

Abstract

Accurate detection of cotton top buds is a key prerequisite for precision topping. However, because of the small size of cotton top buds and their high visual similarity to complex canopy backgrounds, high-efficiency detection in natural fields still faces challenges. To address these challenges, we propose an advanced lightweight, real-time, and accurate detection model, MC-RT-DETR. Specifically, we replace the original ResNet18 backbone network of RT-DETR-r18 with MobileNetV3, which significantly reduces the model parameters and computational complexity. Then, we integrate the C2f module into the feature fusion network, which effectively enhances the multiscale feature representation. We employ Wise-IoU to optimize bounding box regression, achieving high-IoU localization accuracy for small targets. Finally, strategies such as increasing the bounding box regression loss function and using MixUp and Copy-Paste data enhancement for decoupled optimization are combined, which improves the robustness of the model in the context of complex crowns. Experiments show that MC-RT-DETR achieves mAP@50 of 96.00% and mAP@50–95 of 83.59% while reducing the number of parameters and GFLOPs by 47.2% and 57.3%, respectively. Edge deployment on Jetson Orin Nano demonstrates that the model achieves a frame rate of 65.9 FPS, fully satisfying real-time field requirements.

Keywords: precision topping; cotton top bud detection; lightweight model; RT-DETR; edge deployment

Check for updates

Received: 19 June 2026
Revised: 25 July 2026
Accepted: 28 July 2026
Published: 3 August 2026

Correction Statement: This article has been republished with a minor change. The change does not affect the scientific content of the article and further details are available within the backmatter of the website version of this article.

Copyright: © 2026 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license.

1. Introduction

Cotton is an important cash crop, and its production efficiency and cultivation quality are closely related to the global supply of textile raw materials and agricultural economic development [1]. In the growth cycle of cotton, topping is a key agricultural measure to control the top advantage, promote the distribution of nutrients to the reproductive organs, and ultimately maximize the number and yield of cotton bolls [2]. Topping can improve the crown structure, boost the growth of fruiting branches, and lay a solid foundation for high yield. At present, topping is mainly carried out manually, chemically, and mechanically. Traditional manual topping relies heavily on human experience and presents problems such as high labor intensity, low efficiency, and poor operation consistency [3]. Conventional chemical and mechanical topping methods struggle to adapt to the growth status of

individual plants, causing missed targets and chemical waste. With the rising labor cost and the rapid development of intelligent agricultural machinery, the use of machine vision to automatically identify cotton top buds has become an important research frontier. This technology provides location and status information for precision topping, variable-rate chemical regulation, and intelligent robotic execution.

However, cotton top bud detection is essentially a small-target detection task in natural field environments. Cotton top buds are inherently minuscule and irregularly shaped and possess weak textural features. In addition, their color is highly similar to that of the surrounding young leaves, branches, and complex canopy background. In real agricultural images, leaf occlusion, plant overlap, severe light fluctuations, camera distance changes, and complex background interference seriously affect the detection of top buds. Therefore, the visual model must not only determine whether top buds exist but also carry out precise spatial positioning. For mechanical topping or fixed-point spraying applications, even a small deviation in the positioning of the bounding box affects the operation accuracy of the actuator, leading to erroneous cutting, missed targets, or spray drift. Thus, the cotton top bud detection model needs to have high detection accuracy, strong complex environment robustness, and good edge equipment deployment ability at the same time [4].

In recent years, target detection methods based on deep learning have been widely used in the field of agricultural visual perception [5] and have achieved remarkable results in fruit detection, disease detection, and weed identification [6–9]. Object detection architectures are categorized into two-stage detectors, single-stage detectors, and Transformer-based end-to-end models. Typical two-stage detectors, such as Faster R-CNN, generate region proposals via a dedicated network, showing significant advantages in detection accuracy [10]; meanwhile, FPN enhances multiscale feature representation, making it suitable for multiscale targets [11]. However, two-stage detectors are limited by their complex structure, large number of parameters, and slow inference speed, making them difficult to deploy on mobile field platforms or edge devices in real time. Single-stage detectors, such as SSD, realize detection through multiscale feature maps and provide quick reasoning [12]; nonetheless, for small targets in complex contexts, their feature representation is often insufficient. The You Only Look Once (YOLO) series of algorithms has been praised for its compact structure and fast speed and has become the mainstream choice in the field of agricultural target detection [13]. Specifically for cotton top bud detection, existing studies have extensively explored methods such as network compression, small-target feature enhancement, attention mechanism, and multiscale fusion. For tiny and visually unrecognizable top buds in fields, Chen et al. [14] proposed an improved Faster R-CNN integrated with a dynamic mechanism. The model can effectively identify top buds, but the inherent real-time deployment limitations of this two-stage model remain. Song et al. [15] developed a dynamic recognition and positioning method based on improved Cascade R-CNN and combined with RGB-D cameras. Although this method enhances the positioning ability in complex scenarios, its cumbersome model structure and sensor system limit its real-time application in fields. Peng et al. [16] added a small-target detection layer and an attention mechanism in YOLOv5 to improve its accuracy rate, but these additions also increased computational overhead. Zhao et al. [17] proposed the CottonBud-YOLOv5s lightweight cotton top bud detection algorithm, which adopts a ShuffleNetV2 lightweight backbone and a DySample dynamic upsampling module to achieve good balance between detection accuracy and speed. Zu et al. [18] combined the CottonBud-YOLOv5s algorithm with an RGB-D camera to realize real-time dynamic positioning. Models such as Bud-YOLO, DFR-SN-YOLO, and CBLN-YOLO have also promoted the development of cotton top bud detection research to a certain extent from the perspectives of lightweight structure design, feature fusion, attention mechanism, and segmentation detection integration [19–21]. These

studies show that deep learning models have become the main technical route for the visual perception of cotton top buds. However, an obvious limitation still exists: At present, most studies focus on YOLO-based frameworks. Although YOLO's single-stage architecture can achieve rapid target positioning and classification in a forward propagation, its detection results essentially rely on non-maximum suppression (NMS) for postprocessing. In the case of dense crowns, the top buds are gathered, closely adjacent or partially overlapping, and NMS will pose a serious risk of wrongly inhibiting adjacent true positive bounding boxes (pseudo-negative).

The development of end-to-end detectors provides a new way of thinking to solve this problem. Detection Transformer (DETR) pioneered the modeling of object detection as a set prediction problem, utilizing the Hungarian algorithm for bipartite matching to completely eliminate the reliance on anchor designs and NMS postprocessing [22]. The subsequent iteration, deformable DETR [23], alleviates the problems of slow convergence speed of the original DETR and insufficient use of small-target characteristics through the deformable attention mechanism. DINO [24] and other methods further enhance the training stability and detection accuracy through query denoising and improved matching strategies. RT-DETR [25] integrates an efficient hybrid encoder and an uncertainty-minimal query selection mechanism into the DETR framework, achieving excellent real-time performance and demonstrating that DETR models can outperform the YOLO series in accuracy and speed. RT-DETR has the characteristics of end-to-end prediction, no NMS architecture, and global context modeling, showing great potential in intensive target detection in complex contexts. RT-DETR and its variants have been successfully used in agricultural tasks such as tea disease, weed, tomato, and strawberry ripeness detection, demonstrating good application prospects [26–29]. However, the direct utilization of the RT-DETR baseline in field cotton top bud detection faces two practical bottlenecks. First, the parameters and computational footprint of RT-DETR are significantly high, posing computational strain when deployed directly on edge devices with limited computing power. Second, if the model compression is not handled properly, it may weaken the expression of fine-grained characteristics, on which small targets such as top buds depend, thus affecting the detection and positioning performance.

On the basis of the above analysis, this study investigates ways to resolve the trade-off between model compression and performance in the detection of cotton top buds and constructs a lightweight target detection method suitable for complex environments based on the RT-DETR framework. Through a series of optimizations, this study aims to achieve reasonable balance between detection accuracy, positioning fidelity, model complexity, and actual deployment feasibility to provide a reliable visual perception method for precision cotton topping. The main contributions of this study are summarized as follows:

(1) In response to the challenges of small scale, weak texture, and complex background of cotton top buds, we propose a lightweight end-to-end detection model based on RT-DETR suitable for real-time edge deployment. It provides a new research paradigm that is different from the traditional convolutional detection framework.
(2) A decoupled feature extraction and fusion architecture combining MobileNetV3 and C2f is proposed. By synergizing these modules, the framework bounds the computational cost via the lightweight backbone while recovering essential local spatial details through the richer gradient flow of the neck, thereby satisfying the dense feature requirements of the RT-DETR encoder.
(3) A training paradigm tailored for canopy occlusion is constructed. Wise-IoU (WIoU) is coupled with MixUp and Copy-Paste augmentations, and the dynamic focusing mechanism of WIoU is utilized to effectively manage the varying-quality samples generated by the augmentations, stabilizing the bounding box regression process.

(4) Through ablation studies, comparative evaluations against mainstream models, cross-dataset generalization tests (Global Wheat Head Detection [GWHD] dataset), and edge deployment on Jetson Orin Nano (NVIDIA, Santa Clara, CA, USA), we systematically validate the effectiveness of the proposed method in terms of accuracy, complexity, cross-dataset applicability, and practical deployment performance.

2. Materials and Methods

2.1. Dataset

This study used the public cotton top bud detection dataset "Cotton Top Bud," which was published by Song et al. [30] in his YOLOv8-CTB study and is publicly provided through the Kaggle platform. According to the original text, the dataset was collected from the Cotton Technology Research Center of Wudi County, Binzhou City, Shandong Province, which adopts the "three rows and one membrane" planting model. This high-density agronomic practice typically causes dense foliage and frequent mutual occlusion among adjacent cotton plants, thereby ensuring that the dataset realistically reflects the complex visual conditions of actual topping operations. The original resolution is 2160 × 3140 pixels, and the dataset is saved in JPG format. The MakeSense online annotation tool was used for manual annotation, covering various perspectives such as close-range, long-range, and occluded views. To effectively prevent model overfitting, Song et al. applied various data enhancement technologies to the originally collected data. Considering the impact of light changes in the natural environment, they added glare points to some images, adjusted the brightness, and carried out grayscale processing to realistically simulate different natural lighting conditions. At the same time, to simulate the image changes caused by wind disturbance, they adopted image processing operations such as noise addition, random rotation, and mirror flipping.

The public dataset used in this study contains 4353 preprocessed images, which were randomly divided into training, verification, and test sets consisting of 3288, 532, and 533 images, respectively. The top bud target was marked with a rectangular bounding box into a single category. During the model training and testing processes, the size of the input images was uniformly adjusted to 640 × 640 pixels. Figure 1 shows some of the images in the dataset.

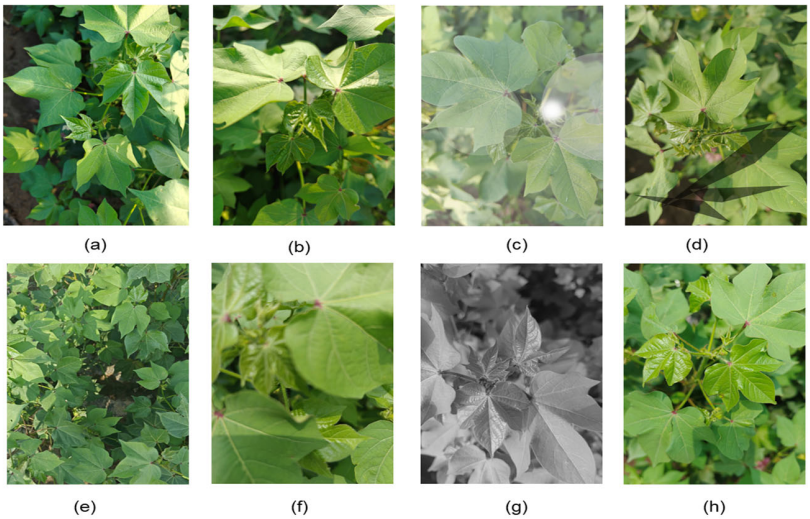


Figure 1. Dataset samples: (**a**) slight obstruction; (**b**) serious obstruction; (**c**) bokeh addition; (**d**) noise addition; (**e**) far distance; (**f**) natural bokeh; (**g**) grayscale conversion; (**h**) negative sample.

2.2. Model Design

2.2.1. RT-DETR Baseline Model

RT-DETR is a real-time, end-to-end object detection model based on the DETR framework. Unlike traditional YOLO detectors, RT-DETR models object detection as a set prediction problem. It eliminates the reliance on NMS postprocessing and directly outputs the final detection results through a decoder. In this study, RT-DETR-r18 was selected as the baseline model. It utilizes ResNet18 as the backbone network, offering favorable balance between detection accuracy and model complexity. However, for the task of in-field cotton top bud detection, RT-DETR-r18 requires further optimization. First, the parameter counts and computational cost of the ResNet18 backbone are relatively high, which is unfavorable for lightweight deployment on resource-constrained platforms. Second, cotton top buds are small targets with weak textures, often accompanied with occlusion and complex background interference, placing high demands on the fine-grained feature extraction capability of the backbone network and the subsequent feature fusion capability. Therefore, this study conducted a synergistic optimization of lightweight design and detection performance based on RT-DETR-r18.

2.2.2. Overall Architecture of the MC-RT-DETR Model

RT-DETR-r18 offers strong detection capabilities, but its deployment on agricultural edge devices is constrained by computational demands. A common practice is replacing the backbone with a lightweight network, yet this approach inherently compresses spatial details. Consequently, it limits the dense feature maps required by the Transformer encoder to establish effective attention mechanisms for small targets (e.g., cotton top buds).

To resolve this limitation, the MC-RT-DETR framework utilizes a synergistic feature extraction and fusion strategy. While MobileNetV3 is employed to bound the computational overhead of the backbone, C2f modules are integrated into the neck to compensate for the resulting feature degradation. This specific configuration recovers local spatial information through richer cross-stage gradient flow branches, ensuring that the Transformer encoder receives sufficient structural details without scaling up the backbone. Additionally, extreme data augmentations generate complex occlusion samples to improve dense canopy detection. The dynamic focusing mechanism of WIoU is utilized to stabilize training under these conditions, preventing the regression loss from being dominated by artificial low-quality bounding boxes. The overall architecture is illustrated in Figure 2.

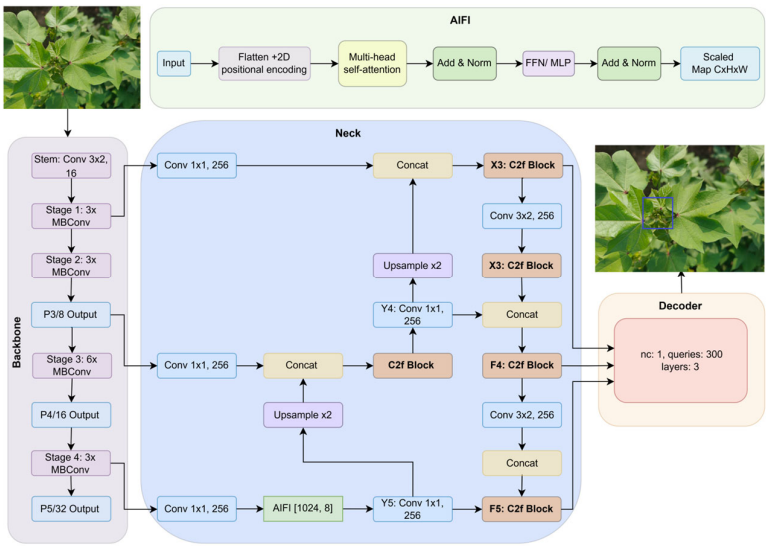


Figure 2. Schematic diagram of the proposed MC-RT-DETR architecture.

2.2.3. MobileNetV3 Lightweight Backbone Network

Cotton topping operations typically need to run in real time on mobile platforms or edge computing devices in fields. Therefore, the detection model must have low computational complexity and high inference speed. For this reason, MobileNetV3 is used in RT-DETR-r18 to replace the original ResNet18 backbone network. MobileNetV3 combines techniques such as deep separable convolutions, a reverse residual structure, the squeeze-and-excitation (SE) attention mechanism, and hardware-aware neural architecture search, enabling it to maintain strong feature extraction capabilities at a low computational cost [31]. Compared with the traditional convolutional backbone, MobileNetV3 has fewer parameters and lower FLOPs. The network structure of MobileNetV3 is shown in Figure 3.

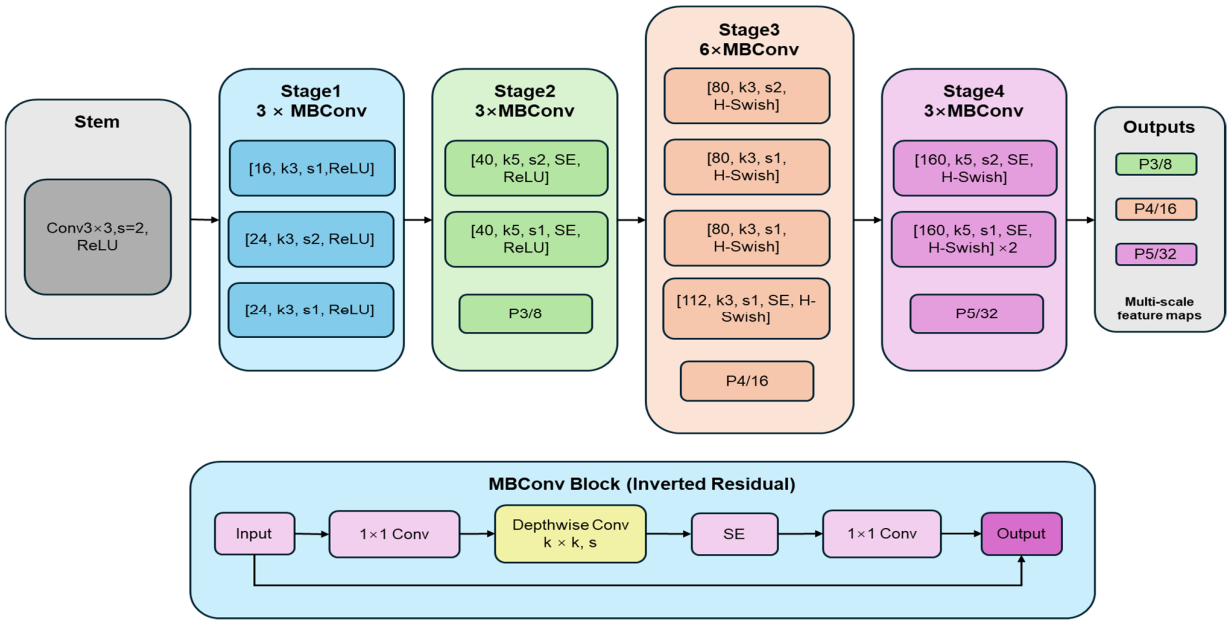


Figure 3. Structure of the MobileNetV3 backbone network.

From a structural composition perspective, MobileNetV3 consists of an initial convolutional layer (stem) and multiple stages that output multiscale features. The stem uses a 3×3 convolution with a step length of 2 for initial sampling. Then, the network enters multiple stages to gradually expand the receptive field and enhance the semantic representation capability [32]. For detection tasks, multiscale feature maps are generally extracted from three stages with strides of 8, 16, and 32 and fed into the subsequent neck for fusion, which is also the key prerequisite for MobileNetV3 to serve as an alternative backbone for RT-DETR. For ease of readability, all major mathematical symbols and their physical definitions used throughout Sections 2.2–2.5 are summarized in Appendix A (Table A1) at the end of this manuscript.

A standard convolution requires simultaneous completion of spatial convolution and channel mixing, and its computational cost is shown as follows:

$$FLOPs_{standard} = C_{in} \times C_{out} \times K^2 \times H \times W \tag{1}$$

By contrast, a depthwise separable convolution splits this process into two steps: First, a depthwise convolution is used to perform spatial filtering independently within each channel. Then, a 1×1 pointwise convolution is adopted for information fusion across channels. Its computational cost can be expressed as

$$FLOPs_{dw} = C_{in} \times K^2 \times H \times W + C_{in} \times C_{out} \times H \times W \tag{2}$$

The reduction ratio of computational cost is shown below:

$$\frac{FLOPs_{depthwise}}{FLOPs_{standard}} = \frac{1}{C_{out}} + \frac{1}{K^2} \tag{3}$$

In addition to its lightweight design, MobileNetV3 also enhances channel representation capability through the SE attention mechanism. The SE module first performs global average pooling on the input feature map to obtain a channel-level descriptor vector. Then, it learns the importance coefficient of each channel through two fully connected mapping layers. Finally, it applies this coefficient to the original feature map to achieve channel recalibration. Its form can be expressed as follows:

$$z = GlobalAvgPool(X) = \frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} X_{i,j} \in R^C \tag{4}$$

$$s = \sigma(\ W_2 \times ReLU(W_1 \times z)) \tag{5}$$

$$X\prime = s\ \odot X \tag{6}$$

where $X \in R^C$ is the input feature map, W_1 and W_2 are learnable parameters, σ is the sigmoid activation function, and $\odot$ denotes element-wise multiplication. For the cotton top bud detection task, when the top bud is similar in color and texture to tender leaves, stem tips, or shadowed areas, the network can strengthen target-related responses and suppress background interference channels through channel reweighting, thereby improving the feature discriminative capability.

From the perspective of stage design, MobileNetV3 embodies a progressive feature organization model from shallow to deep. The front stages adopt fewer channels and higher resolutions, mainly retaining edge, texture, and local morphological information; the middle stages begin to introduce higher channel numbers and stronger semantic abstraction; the rear stages elevate semantic representation through larger dimensions and more bottleneck units containing SE. This structure allows it to maintain lightweight feature extraction while outputting multiscale features suitable for detection tasks. Therefore, replacing ResNet18 with MobileNetV3 can accomplish feature extraction at a decreased backbone cost while preserving multiscale output capabilities.

2.2.4. C2f Feature Fusion Enhancement Module

After the backbone network was replaced with MobileNetV3, the model complexity was significantly reduced, but this approach inevitably led to a decline in the model's multiscale feature representation capability. For small-object detection tasks like cotton top buds, fully fusing shallow spatial details and deep semantic information is crucial. To enhance the feature fusion capability, this study introduced the C2f module in the neck network to replace the original RepC3 module. C2f is a feature fusion module proposed in YOLOv8, which enhances the feature fusion capability by introducing additional cross-layer connections [33]. It strengthens feature reuse and improves the gradient propagation path through channel splitting, cross-stage connections, and feature concatenation. Compared with RepC3, the C2f structure is more flexible and can enhance feature representation capabilities at a lower computational cost. Its network structure is shown in Figure 4.

Given the input feature $\boldsymbol{X} \in \mathrm{R}^{\mathrm{C} \times \mathrm{H} \times \mathrm{W}}$, C2f first performs channel adjustment via a 1×1 convolution and splits the intermediate feature into two parts, as shown as follows:

$$[\boldsymbol{y}_0, \boldsymbol{y}_1] = Split(f_{1\times1}(\boldsymbol{X})) \tag{7}$$

where $f_{1\times1}$ denotes the 1×1 convolution operation, and Split represents the partitioning along the channel dimension. Subsequently, one part of the feature is directly retained as a short branch, while the other part is fed into several bottleneck units to extract deeper features layer by layer, that is,

$$\boldsymbol{y}_i = \mathcal{B}_{i-1}(\boldsymbol{y}_{i-1}), \quad i = 2, 3, \ldots, n+1 \tag{8}$$

where B_{i-1} denotes the $(I-1)$-th bottleneck unit. Ultimately, the initial branch feature and the output of each bottleneck level are concatenated along the channel dimension, and unified fusion is performed through a 1×1 convolution to yield the module output, as indicated below:

$$\boldsymbol{Y} = f_{1\times1}(Concat(\boldsymbol{y}_0, \boldsymbol{y}_1, \ldots, \boldsymbol{y}_{n+1})) \tag{9}$$

From the above process, C2f does not merely conduct a simple concatenation; instead, it retains the initial branch feature and the intermediate features enhanced layer by layer, allowing them to participate in the final fusion. In this way, the module output contains shallow edge and texture information, layer-by-layer enhanced local structure information, and higher-level semantic information, thus improving the richness of feature expression. Its advantages are mainly reflected in two aspects: First, the reuse of features is comprehensive. Given that the intermediate output of each layer is retained and participates in the final integration, the network can reuse existing information at different depths. This advantage is especially important for targets such as cotton top buds because they are small in size and weak in texture and their local features are easily disturbed by the background. Second, the gradient propagation path is rich. Multibranch connection enables reverse propagation to reach the front layer in a short path, which helps alleviate the common feature attenuation problem in lightweight networks and improves the stability of model training.

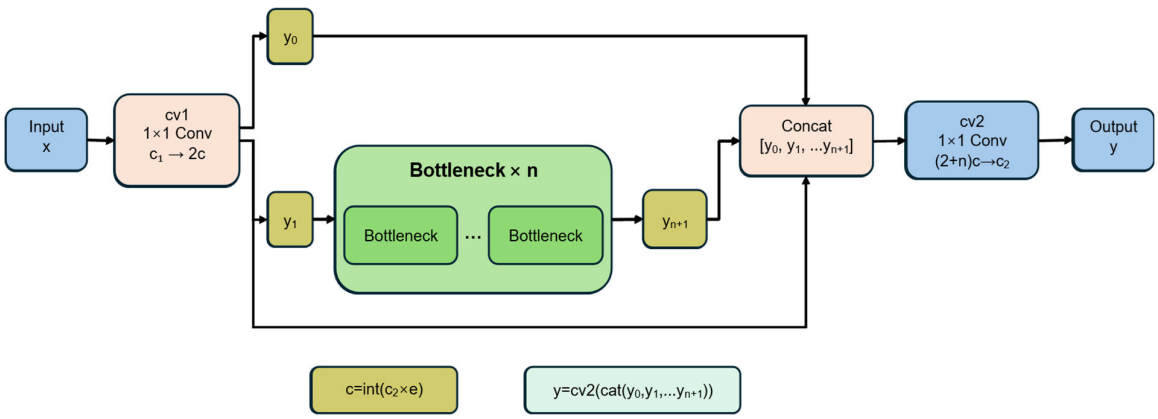


Figure 4. Structure of the C2f module.

For the task in this study, this structure is conducive to retaining the edge, tip morphology, and local texture details of the top buds, thereby mitigating missed detections and localization deviations that might arise from purely substituting a lightweight backbone. By contrast, RepC3 places emphasis on improving inference efficiency through the RepConv structure. This structure actually involves three steps: performing convolutional transformations, extracting features through the main branch, and fusing them in a residual manner. RepC3 has advantages at the engineering deployment level; however, in the cotton top bud detection scenario in this study, because the ResNet18 backbone is replaced with the lighter MobileNetV3, the neck part needs to undertake a stronger fine-grained feature compensation task. Under these circumstances, C2f is more suitable to be embedded into

the improved RT-DETR as a feature enhancement module owing to its more sufficient retention of intermediate features and denser fusion paths.

2.3. Optimization of the Training Strategy for the Lightweight Model

Original large-parameter models like RT-DETR typically achieve rapid convergence with a relatively short training period (e.g., 200 epochs). In this study, after the backbone network was replaced with MobileNetV3, the model's parameter count was substantially reduced. Although lightweight networks possess fast inference speeds, their fitting capability in the early training stages is relatively weak. Employing the traditional short-period linear decay strategy easily causes the model to fall into suboptimal solutions, failing to fully unleash its feature extraction potential [34]. Therefore, this study made targeted adjustments to the training strategy: First, the total number of training epochs was expanded from 200 to 400 of the benchmarks to provide sufficient parameter update space for the model. Second, the traditional linear learning rate attenuation was abandoned, and the cosine annealing learning rate scheduling strategy was adopted instead. The initial learning rate was set to 0.0004, and it was smoothly reduced to 0.5% of the initial value (lrf = 0.005) in the later stage of training. To avoid gradient oscillation in the lightweight backbone network at the beginning of training, this study increased the number of pretraining rounds from 3 rounds to 10 rounds. This series of pretraining strategies ensured the fine-grained convergence of the improved lightweight model in subsequent training, laying the foundation for high-precision positioning.

2.4. WIoU Bounding Box Regression Optimization

For the cotton top bud detection task, the positioning accuracy of the bounding box is crucial. In precision topping, the detection box not only determines whether the target is successfully identified but also provides spatial location information for subsequent top position planning or mechanical operation. Therefore, apart from detecting the target, improving the matching degree between the detection box and the actual top bud region is equally critical. The original RT-DETR-r18 model primarily employs the default GIoU-like loss for bounding box regression. Given the predicted box and the ground-truth box as B and B^{gt}, respectively, IoU can be expressed as

$$IoU = \frac{\left|B \cap B^{gt}\right|}{\left|B \cup B^{gt}\right|} \tag{10}$$

Building upon this expression, GIoU further considers the outer enclosing area of the predicted box and the ground-truth box, which is defined as

$$GIoU = IoU - \frac{\left|C \backslash \left(B \cup B^{gt}\right)\right|}{|C|} \tag{11}$$

where C represents the smallest enclosing rectangle covering B and B^{gt}. The corresponding GIoU loss is

$$\mathcal{L}_{GIoU} = 1 - GIoU \tag{12}$$

Although GIoU can account for the spatial relationship between the predicted box and the ground-truth box, in the complex cotton canopy background, samples of different qualities do not contribute equally to the training. For instance, some severely occluded or background-interfered samples may produce low-quality predicted boxes. If excessively strong constraints are imposed on such samples during training, it may affect the overall convergence of the model. To address this issue, this study introduced the WIoU loss

function. WIoU incorporates a dynamic focusing mechanism based on the fundamental IoU loss, which can be written as

$$\mathcal{L}_{WIoU} = r(\beta) \cdot (1 - IoU) \tag{13}$$

where $r(\beta)$ represents the dynamic focusing factor related to sample quality, and β is used to characterize the prediction quality or outlier degree of the current sample. With reference to the design philosophy of WIoU, it can be expressed as

$$\beta = \frac{\mathcal{L}^*_{IoU}}{\overline{\mathcal{L}_{IoU}}}, \quad \mathcal{L}^*_{IoU} = 1 - IoU \tag{14}$$

where $\overline{\mathcal{L}_{IoU}}$ represents the moving average of the IoU loss during the training process. By applying differentiated gradient weights to samples of varying quality, this mechanism enables the model to reduce its excessive focus on extremely low-quality abnormal samples and concentrate more on samples with valid localization errors. For cotton top bud detection, this strategy contributes to improving the bounding box regression quality under complex backgrounds, local occlusions, and weak texture conditions, thereby enhancing target localization accuracy.

2.5. Decoupled Optimization Strategy

Preliminary experiments indicated that after the introduction of WIoU, the model's high-IoU localization capability was improved, but the recall rate experienced a slight decrease. This finding demonstrated that while stronger localization constraints improved box regression quality, they might render the model's predictions more conservative for blurry, occluded, or low-confidence targets. Missed detections in the cotton top bud detection task must also be minimized. To this end, this study adopted a decoupled optimization strategy, conducting synergistic optimization from two aspects: loss function adjustment and data augmentation.

First, under the premise of maintaining the overall training framework unchanged, the weight of the regression-related loss terms was appropriately increased to strengthen localization learning. Its overall form can be written as

$$\mathcal{L} = \mathcal{L}_{cls} + \lambda_{reg}\mathcal{L}_{reg}, \quad \lambda_{reg} > \lambda^{base}_{reg} \tag{15}$$

where $\mathcal{L}_{cls}$ denotes the classification loss. $\mathcal{L}_{reg}$ represents the bounding box regression-related loss term; in this study, the box quality constraint was implemented using WIoU. λ_{reg} is the regression term weight. λ^{base}_{reg} is the corresponding setting of the baseline model. Increasing the regression term weight can reinforce the model's learning of target geometric position and bounding box matching relationships during the training phase. Second, to enhance the model's adaptability to occlusions and complex backgrounds, this study introduced MixUp and Copy-Paste data augmentation strategies. For MixUp, given two training images and their annotations as (X_a, Y_a) and (X_b, Y_b), the mixed sample can be expressed as

$$\widetilde{X} = \lambda X_a + (1 - \lambda)X_b, \ \lambda \sim Beta(\alpha, \alpha), \ \widetilde{Y} = Y_a \cup Y_b \tag{16}$$

where λ is the mixing coefficient, and Beta(α, α) represents the beta distribution. MixUp improves the distribution continuity and background diversity of the training sample by linearly fusing two images, thus enhancing the generalization ability of the model. For Copy-Paste, the target image is let be X_t, the source image be X_s, the binary mask corresponding to the target region to be pasted be M, and the geometric transformation

operation be $T(\cdot)$. Then, the enhanced image and its corresponding label updates can be expressed as

$$X_{cp} = (1 - M) \odot X_t + M \odot T(X_s), \quad Y_{cp} = Y_t \cup T(Y_s) \tag{17}$$

MC-RT-DETR makes targeted adjustments to the default hyperparameters. On the loss side, the bounding box regression loss function is increased from the default of 7.5 to 10.0 to force the model to focus on the high-precision localization gradients provided by WIoU during the later stages of training; the classification loss function remains unchanged at 0.5. On the data augmentation side, MixUp is set to 0.15 and Copy-Paste to 0.3. A MixUp probability of 0.15 increases the integration of complex field backgrounds without excessively disrupting the semantic structure of the original background space; meanwhile, a Copy-Paste probability of 0.3 forcibly copies and pastes cotton top bud instances to different locations, simulating local occlusions and multiscale variations under high-density planting. The two techniques complement each other, enriching the feature space of hard-to-detect samples. For the cotton top bud task, this study disables the setting that typically turns off mosaic augmentation during the final stage of conventional training (i.e., setting "close_mosaic = 0"), ensuring that strong data augmentation is applied throughout all 400 training epochs. This approach maximally forces the model to learn robust features of top buds within extremely complex blended backgrounds, thereby effectively addressing the false negatives caused by relying solely on WIoU.

2.6. Evaluation Metrics

This study used precision, recall, mAP50, mAP50-95, parameters, FLOPs, FPS, and other indicators to comprehensively evaluate the performance of the proposed model.

Precision indicates the proportion among all targets predicted to be cotton top buds that are really top buds; recall represents the ratio of cotton top buds correctly detected; mAP50 indicates the average accuracy rate when the IoU threshold is 0.50; mAP50-95 implies that the average mAP of the IoU threshold ranges from 0.50 to 0.95 (the step length is 0.05). The calculations for the evaluation indices are shown as follows:

$$Precision = \frac{TP}{(TP + FP)} \tag{18}$$

$$Recall = \frac{TP}{(TP + FN)} \tag{19}$$

$$mAP_{50} = \frac{1}{n}\sum_{i=1}^{n} AP_{50}^{(i)} \tag{20}$$

$$mAP50 - 95 = \frac{1}{10}\sum_{i=0}^{9} AP_{0.5+0.05i} \tag{21}$$

TP, FP, and FN denote true positives, false positives, and false negatives, respectively.

3. Results

3.1. Experimental Environment Configuration

The training, verification, and testing of all models in this study were carried out on the same experimental condition to ensure the rigor and reproducibility of the experiments. The input image size was uniformly adjusted to 640 × 640 pixels, and the batch size was set to 16. The model training was conducted on a workstation running the Ubuntu 22.04 operating system (Canonical Ltd., London, UK), powered by an NVIDIA GeForce RTX 3090 GPU (NVIDIA, Santa Clara, CA, USA). The deep learning framework adopted was

PyTorch version 2.0.0. Further specific hardware and software environment configurations are detailed in Table 1.

Table 1. Training environment configurations.

Item	Configuration
Operating system	Ubtu 22.04
GPU	NVIDIA GeForce RTX 3090
VRAM	24 GB
Deep learning framework	PyTorch 2.0.0
CUDA version	11.8
Python version	3.8.20

3.2. Experimental Parameter Settings

In the comparative and ablation experiments, the training epochs was carefully configured to ensure complete convergence and a strictly fair evaluation. Traditional classic models (such as Faster R-CNN and SSD) were trained for 200 epochs. For the proposed MC-RT-DETR, considering that it incorporates substantial lightweight structural modifications and robust data augmentation strategies, the training cycle was extended to 400 epochs to ensure the model could fully learn the deep features of the targets. To ensure a fair comparison, the baseline model (RT-DETR-R18) and all YOLO series models were also trained from scratch for 400 epochs. The specific training hyperparameters are listed in Table 2.

Table 2. Hyperparameter settings for the MC-RT-DETR model.

Parameter	Value	Parameter	Value
Epochs	400	Optimizer	AdamW
Batch size	16	Initial learning rate (lr0)	0.0004
Image size	640 × 640	Learning rate decay factor	0.005
Learning rate scheduling	Cosine annealing	Weight decay	0.0005
Basic data augmentation	Mosaic (1.0)	MixUp/Copy-Paste	0.15/0.30

3.3. Comparative and Ablation Experiments

3.3.1. Comparison of Different Backbone Networks

To justify the selection of MobileNetV3 as the backbone network, this study conducted a comparative analysis by introducing four mainstream lightweight architectures (GhostNet, StarNet, RepViT, and MobileNetV3) into the baseline model. The results are presented in Table 3.

Table 3. Performance evaluation of various backbone architectures.

Backbone	Parameters/M	FLOPs/G	P/%	R/%	mAP50/%	mAP50-95/%
ResNet18	20.84	57.9	96.25	91.39	96.15	82.93
GhostNet	11.45	28.5	91.50	88.62	93.10	74.85
StarNet	12.80	32.4	91.88	89.20	93.45	75.12
RepViT	13.92	34.6	92.15	89.55	93.80	75.90
MobileNetV3	12.33	31.3	92.61	90.04	94.28	76.93

As shown in Table 3, although ResNet18 achieved the highest detection accuracy, its massive parameter count (20.84 M) and computational overhead somewhat restrict its deployment on edge devices such as agricultural drones or robotic arms. After it was

substituted with lightweight networks, the parameters of all models dropped significantly, but the performance varied noticeably.

GhostNet, despite having the lowest parameter count (11.45 M), suffered from insufficient feature extraction capability, possibly because its cheap linear transformations generated redundant features that were inadequate to preserve fine-grained textures and edge information of small-sized cotton top buds, causing its mAP50-95 to plummet to 74.85% and making it unsuitable for high-precision localization. StarNet and RepViT, as newer architectures, performed excellently on general datasets but showed limited generalization in agricultural scenarios with complex background interference and small targets like cotton top buds. This limitation can be attributed to their design emphasis on global semantic modeling rather than local detail perception, making them less effective for accurately localizing tiny buds under cluttered field conditions. In particular, they achieved mAP50 scores of only 93.45% and 93.80%, respectively, with significantly reduced mAP50-95. By contrast, MobileNetV3 demonstrated the optimal performance balance. Integrating the SE attention mechanism into the inverted residual structure of MobileNetV3 forced the network to dynamically assign higher weights to key feature channels of the cotton top buds, effectively suppressing field background noise. This channel-wise recalibration is especially beneficial for small-target detection, given that buds occupy only a few pixels in feature maps. Moreover, the SE module can highlight the most relevant channels, while the inverted residual structure maintains relatively high-resolution feature maps in early stages, preserving spatial details critical for accurate localization. MobileNetV3 not only reduced parameters by 40.8% compared with the baseline but also maintained the highest mAP50 (94.28%) and mAP50-95 (76.93%) among all lightweight schemes. Therefore, MobileNetV3 was selected as the optimal backbone network.

3.3.2. Comparison of Different Loss Functions

The lightweight modification of the backbone inevitably caused a degradation in bounding box regression accuracy (mAP50-95). To mitigate this defect, we compared several mainstream improved loss functions on the lightweight model to identify the most suitable optimization strategy for high-precision top bud detection. The results are shown in Table 4.

Table 4. Performance evaluation of various loss functions.

Loss Function/Strategy	mAP50 (%)	mAP50-95 (%)	P (%)	R (%)
GIoU	95.81	80.97	96.89	88.24
Inner_CloU	94.76	75.63	94.06	87.41
Inner_WIoU	95.20	74.70	93.8	88.8
WIoU_Focal	93.73	74.71	93.50	88.79
Focaler_GIoU	93.70	68.60	92.8	89.7
WIoU	93.68	83.05	96.53	87.72

Experiments revealed that the attempts to improve the network's recall for small targets by modifying the penalty mechanism of the loss function (e.g., introducing a focal mechanism or Inner_IoU auxiliary bounding box computation) resulted in severe localization degradation. For instance, while Focaler_GIoU slightly improved recall, its mAP50-95 sharply dropped to 68.60%. In complex agricultural images, aggressive loss functions tend to misclassify background noise as targets (a surge in false positives), thereby corrupting the otherwise clean regression gradients. Conversely, the pure WIoU, through its dynamic nonmonotonic focusing mechanism, perfectly concentrated the network's attention on anchor boxes of ordinary quality, greatly enhancing bounding box fitness and

pushing mAP50-95 to 83.05%. Taking all factors into consideration, this study determined WIoU as the optimal regression loss function. However, because WIoU's predictions are conservative, the model's miss rate slightly increased (recall dropped to 87.72%). To address this trade-off between high-precision regression and high recall, we implemented a decoupled optimization strategy in the subsequent ablation experiments.

3.3.3. Ablation Experiment Results

Ablation experiments were conducted on the cotton top bud dataset using RT-DETR-r18 as the baseline to verify the effectiveness of each proposed improvement. We sequentially analyzed the impact of the MobileNetV3 backbone replacement, the C2f feature fusion module, the training epoch adjustment, the WIoU loss function, and the final decoupled optimization strategy. The results are summarized in Table 5.

Table 5. Results of the ablation study.

Model	Improvement	Parameters (M)	FLOPs (G)	mAP50 (%)	mAP50-95 (%)	P (%)	R (%)
RT-DETR-r18	_	20.84	57.9	96.39	84.01	95.24	92.58
M1	MobileNetV3	12.33	31.3	94.28	76.93	92.61	90.04
M2	M1 + C2f	11.01	24.7	96.20	77.53	93.24	90.74
M3	M2 + 400 ep	11.01	24.7	95.81	80.97	96.89	88.24
M4	M3 + WIoU	11.01	24.7	96.08	83.05	96.53	87.72
MC-RT-DETR	M4 + aug	11.01	24.7	96.00	83.59	95.69	90.39
MC-RT-DETR (4-seed mean) [1]	M4 + aug	11.01	24.7	95.34 ± 0.88	82.75 ± 0.66	95.09 ± 1.09	89.84 ± 0.87

[1] Mean ± std over seeds 0, 42, 123, 3407.

As shown in Table 5, after the original ResNet18 backbone network of the RT-DETR-r18 model was replaced with MobileNetV3, the number of parameters decreased from 20.84 M to 12.33 M, and the computational load decreased from 57.9 G to 31.3 G, indicating that the backbone network compression significantly reduced the complexity of the model. However, mAP50-95 decreased from 84.01% to 76.93%, implying that direct compression weakens the characteristic representation and positioning ability of small top bud targets.

After the introduction of the C2f module, the number of parameters and the computational load were further reduced to 11.01 M and 24.7 G, respectively, while mAP50 increased from 94.28% (M1) to 96.20%. These results showed that the C2f module enhanced feature fusion and partially compensated for the feature loss caused by compressing backbone networks, restoring the detection accuracy to the baseline level.

After the implementation of WIoU, mAP50-95 increased to 83.05%, proving that WIoU can effectively improve the high-IoU positioning performance of the top bud detection frame. However, the recall rate dropped to 87.72%, which means that stronger positioning constraints make the model more conservative in predicting partially obscured or blurred targets.

The transition from M4 to the final model spanned multiple dimensions, including loss function adjustments and data augmentation. To clearly investigate the independent contributions and interactions of each subcomponent in the decoupled optimization strategy, we conducted component-level ablation experiments using M4 as the baseline. The results are shown in Table 6. Increasing the bounding box regression weights alone (M4-a) raised mAP50-95 to 84.03%, indicating that the model's ability to locate high-quality detection boxes was further enhanced. However, this enhancement came at the cost of more stringent predictions, with recall dropping to 87.54%. After MixUp (M4-b) and Copy-Paste (M4-c) were introduced individually, the model was exposed to more complex local occlusions and field-blended backgrounds, and the recall rate rebounded to 90.27% and 90.57%, respectively. Nonetheless, strong data augmentation compromised the original

image clarity, making bounding box regression more challenging. Under the default regression weights, mAP50 declined by approximately 1% compared with that in M4. This phenomenon demonstrates the necessity of a decoupled optimization strategy. The final model combined strong data augmentation with high regression weights, leveraging data augmentation to boost recall while using the increased regression weights to ensure the quality of bounding box localization in complex augmented backgrounds. The synergy between these two approaches enabled the final model to maintain a recall of 90.39% at seed 0, with mAP50 rebounding to 96.00% and an mAP50-95 of 83.59%.

Table 6. Component-level ablation of the decoupled optimization strategy.

Config	Description	P (%)	R (%)	mAP50 (%)	mAP50-95 (%)
M4	WIoU only	96.53	87.72	96.08	83.05
M4-a	M4 + box weight = 10	97.78	87.54	94.36	84.03
M4-b	M4 + MixUp (0.15)	97.50	90.27	94.87	83.00
M4-c	M4 + Copy-Paste (0.3)	95.46	90.57	94.70	83.52
MC-RT-DETR	M4 + all three combined	95.69	90.39	96.00	83.59

Compared with that of the baseline, the recall of the final model decreased by approximately 2.19 percentage points under a single seed and by 2.17 percentage points under multiseed evaluation. In actual agricultural top bud detection and topping operations, missed detections may lead to a decrease in topping coverage. The decline in recall rate observed in this study primarily stemmed from the stricter localization constraints introduced by WIoU, causing the model to adopt a more conservative approach when predicting partially occluded or ambiguous targets. Meanwhile, the fact that mAP50-95 remained stable or even slightly improved indicated that the retained detection boxes were of higher overall quality. In consideration of the strict computational constraints of edge devices and the significant compression benefits provided by this model, this trade-off is acceptable from an engineering perspective.

For further validation of the stability of the results, the last row of Table 5 shows the mean values of MC-RT-DETR with four random seeds. The four-seed mean for mAP50-95 was 82.75 $\pm$ 0.66%, which was on par with the multiseed results of the baseline after 400 training iterations (82.35 $\pm$ 1.71%), while the variance was reduced by approximately 2.6 times. This result indicates that no statistically significant difference in mAP50-95 was detected between the proposed model and the baseline, while MC-RT-DETR exhibited more stable training behavior across seeds. The corresponding significance test analysis is presented in Section 3.3.4.

Finally, we evaluated the computational overhead of each improvement strategy from the perspective of practical deployment. The calculation of the WIoU dynamic loss, the data augmentation via MixUp and Copy-Paste, and the increase in training iterations from 200 to 400 all raised the initial training cost and the time required per training iteration. Nevertheless, all of these additional costs occurred during the offline training phase; during online inference, the data augmentation module and the loss function computation were removed, and the final deployed MC-RT-DETR retained only the lightweight weights corresponding to the MobileNetV3 backbone and the C2f fusion module. Consequently, the final model's parameter count was reduced by 47.1%, and its computational load decreased by 57.3%. This finding indicates that the improvement strategy proposed in this paper achieves good balance between detection accuracy and efficiency during inference at the cost of increased offline training costs, making it valuable for practical engineering deployment.

3.3.4. Analysis of Statistical Significance

To rigorously assess whether the performance differences between MC-RT-DETR and the baseline are statistically significant, this study trained the RT-DETR-r18 baseline and MC-RT-DETR using four random seeds (0, 42, 123, and 3407) under the same training configuration. Both models were trained for 400 epochs with a batch size of 16 and the AdamW optimizer, without using pretrained weights. The data split and hardware environment remained consistent. The results, including mean, standard deviation, 95% confidence intervals, and Welch's *t*-test results for each random seed, are summarized in Table 7.

Table 7. Multiseed comparison between RT-DETR-r18 and MC-RT-DETR.

Metric (%)	RT-DETR-r18	MC-RT-DETR	Δ	*t*	*p*
P	95.63 ± 1.23	95.34 ± 0.88	−0.29	−0.38	0.716
R	92.01 ± 0.70	89.84 ± 0.87	−2.17	−3.86	0.010
mAP50	95.84 ± 0.45	95.09 ± 1.09	−0.75	−1.28	0.256
mAP50-95	82.35 ± 1.71	82.75 ± 0.66	+0.40	+0.44	0.678

Values are mean ± sample std over four random seeds (0, 42, 123, 3407). Δ denotes the mean difference (MC-RT-DETR − RT-DETR-R18). *p*-values are from Welch's two-sample *t*-test. Per-seed results are provided in Appendix B.

MC-RT-DETR achieved a mean mAP50-95 of 82.75% across four seeds, marginally outperforming the baseline with 82.35%, with a mean difference of +0.40 percentage points (Welch's $t = 0.44$, $p = 0.68$).

In terms of localization accuracy, no statistically significant difference was detected between the two models, indicating that the proposed lightweight architecture can achieve detection accuracy comparable to the baseline while reducing the number of parameters by 47.1% and computational complexity by 57.3%. Additionally, the cross-seed standard deviation of MC-RT-DETR decreased from 1.71 for the baseline to 0.66, representing a reduction in variance by approximately 2.6 times and reflecting more stable training behavior, which is of practical significance for reproducible edge deployment.

In conclusion, MC-RT-DETR achieves significant reductions in parameters and computational complexity while maintaining detection accuracy equivalent to the baseline, and it demonstrates better training stability on the cotton top bud detection task. Its performance improvements are supported by a reproducible statistical foundation.

3.3.5. Comparison and Analysis of Different Models

To further validate the performance of the proposed model, we conducted comparative experiments on the test set against mainstream YOLO series models (YOLOv5s, YOLOv8s, YOLOv10s, and YOLO11s), RT-DETR-v2, and the RT-DETR-r18 baseline model. All compared models were set to have a similar parameter scale and evaluated under the same dataset split and experimental environment to ensure fairness. The evaluation results are detailed in Table 8.

SSD performed poorly on the typical small-target task of cotton top bud detection despite its simple inference structure. The recall rate (72.06%) and high-IoU positioning accuracy (with mAP50-95 of only 68.65%) of SSD models were lowest among all models, indicating that it is difficult for shallow networks that lack deep multiscale feature fusion to handle complex canopy backgrounds. As the benchmark of two-stage detection models, Faster R-CNN achieved 95.95% of mAP50 with its region proposal network. However, its huge number of parameters (about 3.8 times that of our model) and computational costs (more than 200 G of FLOPs) severely limit its deployment potential on edge devices. More importantly, despite the complex structure of Faster R-CNN, its recall rate (85.62%) and

mAP50-95 (82.31%) are still behind those of the lightweight model we proposed. Hence, in dense agricultural object detection, simply increasing the depth and complexity of the model cannot effectively solve the problem of missed inspection caused by overlapping.

Table 8. Results of the comparison with mainstream detection models.

Model	Parameters/M	FLOPs/G	mAP50/%	mAP50-95/%	P (%)	R (%)
SSD300	24.4	61.5	95.42	68.65	90.72	72.06
Faster R-CNN	41.5	215.0	95.95	82.31	95.13	85.62
YOLOv5s	9.11	23.8	96.66	90.82	92.35	90.27
YOLOv8s	11.13	28.4	95.41	90.17	92.69	88.03
YOLOv10s	7.22	21.4	93.25	88.16	86.63	85.33
YOLO11s	9.41	21.3	93.41	86.60	87.04	88.79
RT-DETRv2	20.95	58.0	96.29	82.57	97.34	92.35
RT-DETR-r18	20.84	57.9	96.39	84.01	95.24	92.58
MC-RT-DETR	11.01	24.7	96.00	83.59	95.69	90.39

The NMS postprocessing relied upon by YOLO models is highly prone to misjudging and filtering out spatially adjacent genuine buds, inducing a sharp rise in missed detections. YOLO11s and the NMS-free model YOLOv10s not only failed to surpass classic YOLO models on this dataset, but their precision also plummeted to below 88%. RT-DETR-v2 performed quite well on metrics such as mAP50, precision, and recall, but its computational complexity and parameters were essentially the same as those of RT-DETR-r18, making it less suitable for edge devices. By contrast, our model achieved good balance in precision (95.69%), recall (90.39%), and mAP50-95 (83.59%).

To intuitively display the balance between detection accuracy and lightweight degree for each model, this study plotted accuracy–parameter and accuracy–computation bubble charts for mainstream lightweight models and MC-RT-DETR, as shown in Figure 5.

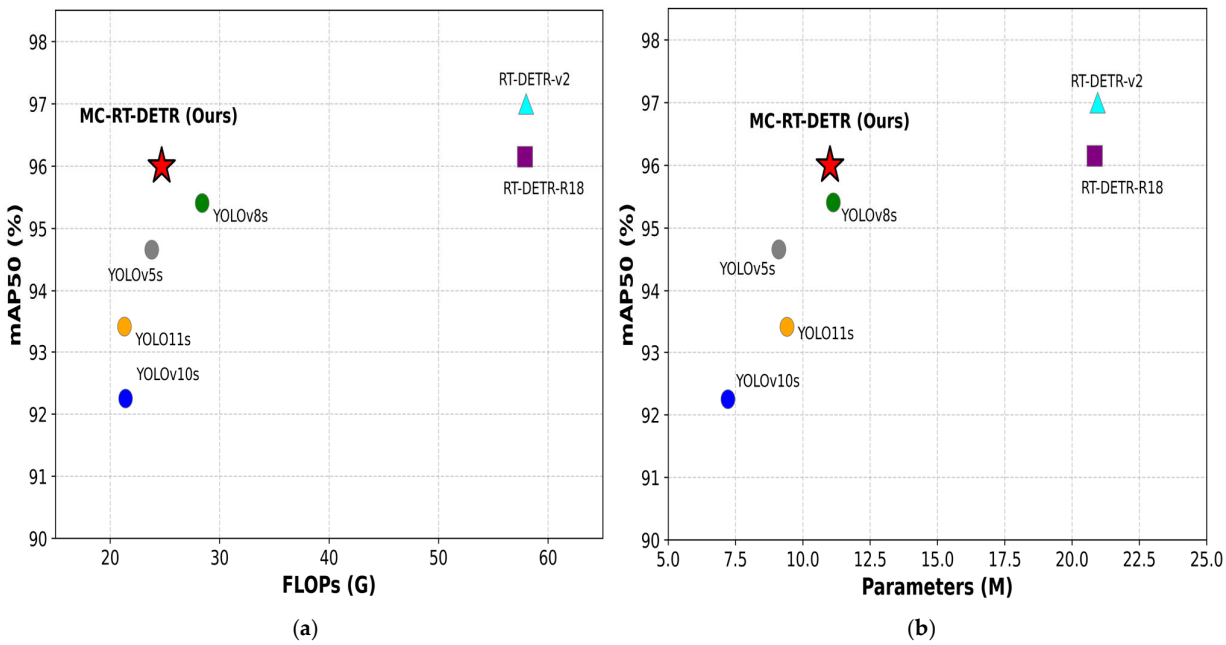


Figure 5. Accuracy–parameter and accuracy–FLOPs trade-offs for the compared models: (**a**) accuracy vs. parameters; (**b**) accuracy vs. FLOPs.

Models located in the upper-left quadrant of the chart possessed high accuracy alongside low model complexity, representing higher comprehensive cost-effectiveness.

Compared with the RT-DETR-r18 baseline and RT-DETR-v2, the proposed MC-RT-DETR achieved a substantial leftward shift (i.e., significantly reduced parameters and computations) while maintaining exceptional detection accuracy. Although RT-DETRv2 has been upgraded in terms of multiscale feature sampling, its parameter count (20.95 M) and computational cost (58.0 G) remained substantial, and its mAP50 (96.29%) on the current dataset did not show a substantial improvement over other models. By contrast, MC-RT-DETR achieved detection performance on par with the leading models while using only about half the computational resources. Furthermore, compared with mainstream lightweight YOLO models (e.g., YOLOv10s and YOLO11s), although MC-RT-DETR had a slightly higher parameter count, it secured lead in mAP50 accuracy (nearly 2–3 percentage points higher). Given the complex background and small targets in the cotton top bud detection task, this increase in complexity within the controllable range paid to obtain high accuracy is completely reasonable and necessary. Overall, MC-RT-DETR achieved the best balance between precision and lightweight design.

3.4. Precision–Recall (PR) Curve Analysis

To visually evaluate the comprehensive performance differences of models in the cotton top bud detection task, we extracted the prediction results of each model on the test set and drew a PR curve, as shown in Figure 6. The PR curve reflects the dynamic trade-off between accuracy and recall rates under different confidence thresholds. The closer the curve is to the upper-right corner of the coordinate axis, the larger the area under the curve (i.e., mAP50), indicating that the model has better comprehensive detection performance.

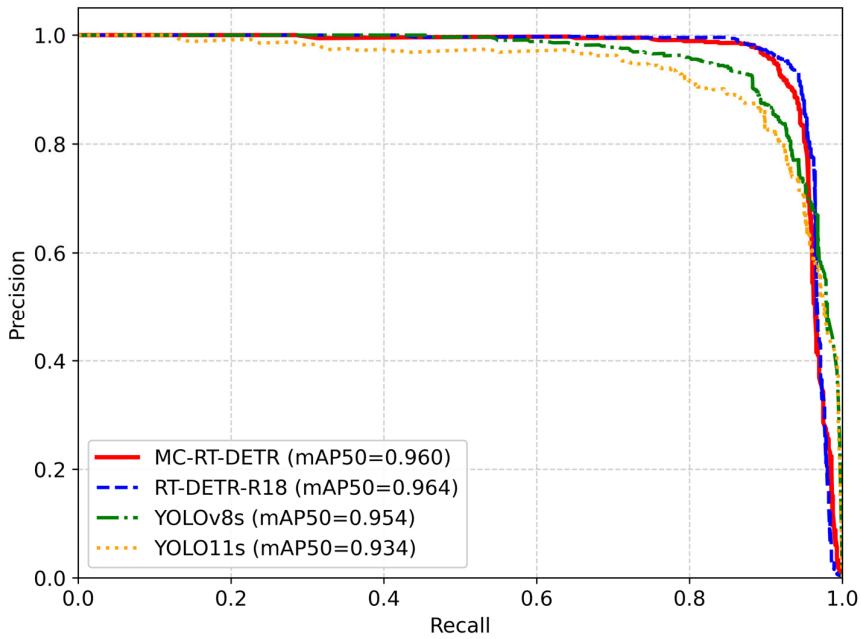


Figure 6. PR curve.

Several key conclusions can be drawn from the curve trends shown in Figure 6. The red solid line, representing the improved model proposed in this paper, almost completely overlaps with the RT-DETR-r18 baseline model (blue thick dashed line), which had a large number of parameters; the mAP50 values of the two models were extremely close (0.960 and 0.964, respectively). This outcome fully demonstrates that the lightweight backbone network and C2f feature fusion strategy introduced in this paper effectively preserve key semantic information while eliminating redundant parameters. More importantly, our model exhibited significantly greater robustness in the high recall range. Comparison

with the YOLO series of models revealed that YOLOv8s (green dotted line) and YOLO11s (orange dashed line) began to experience a drop in precision once recall reached 0.8. This drop indicates that YOLO models inevitably introduce a large number of background false positives when attempting to detect more marginal or challenging samples (such as severely occluded top buds). By contrast, the Transformer-based improved model presented in this paper—owing to the long-range dependencies established by global attention and the strengthened constraints on high-quality anchor boxes imposed by WIoU in the decoupled optimization strategy—maintained its red curve close to the top even as recall approached 0.9 or 0.95, demonstrating exceptional resistance to interference from complex field backgrounds. In particular, not only the PR curve of the latest architecture, YOLO11s, lies below the other three curves, but its mAP50 (0.934) was also lowest among the four. This phenomenon confirms that directly applying network architectures designed for general-purpose large-scale datasets to agricultural small-target scenarios—which feature relatively homogeneous characteristics and dense distributions—is highly likely to cause spatial matching discrepancies. Therefore, optimizing model architectures specific for agricultural scenarios is essential.

3.5. Generalization Validation on the GWHD Dataset

To investigate the structural generalization ability of the MC-RT-DETR model for cross-domain agricultural dense object detection tasks, we performed training and validation from scratch on the GWHD dataset [35]. The GWHD dataset is known for its extremely high object density, severe mutual occlusion, and variable lighting conditions. To achieve a fair comparison of the experimental results and avoid the performance biases caused by differences in training cycles, we set hyperparameter configurations consistent with those of MC-RT-DETR for all comparison models. None of the models used pretrained weights; all were trained uniformly for 400 epochs under the same software and hardware environments, using the AdamW optimizer. The experimental results are shown in Table 9.

Table 9. Generalization validation results on the GWHD dataset.

Model	mAP50 (%)	mAP50-95 (%)	P (%)	R (%)	Parameters/M	FLOPs/G
YOLOv8s	93.5	54.1	91.5	88.4	11.1	28.6
YOLOv10s	92.6	53.4	89.8	87.9	8.1	24.8
RT-DETR-r18	92.3	53.2	90.4	88.8	20.8	57.9
MC-RT-DETR	92.1	53.2	90.2	88.1	11.0	24.7

Experiments demonstrated that MC-RT-DETR exhibited exceptional robustness and computational efficiency even when faced with dense wheat scenes, which present significant feature differences from the source task (cotton top bud detection). Compared with the RT-DETR-r18 baseline model, which also uses a Transformer architecture, our model featured a 47.1% reduction in the number of parameters (11.0 M vs. 20.8 M) and a sharp 57.3% reduction in FLOPs (24.7 G vs. 57.9 G). It not only matched the baseline exactly on the stringent mAP50-95 metric (53.2%) but also showed a marginal 0.2% difference in mAP50 (92.1% vs. 92.3%).

The comparison with mainstream lightweight CNN models (YOLOv8s and YOLOv10s) showed that while YOLOv8s held a slight edge in various accuracy metrics, MC-RT-DETR successfully compressed the originally large and computationally intensive DETR architecture to a computational complexity comparable to that of the lightweight YOLO series (approximately 25 GFLOPs). Moreover, it retained the inherent advantages of the Transformer architecture in global context awareness and end-to-end detection (without NMS).

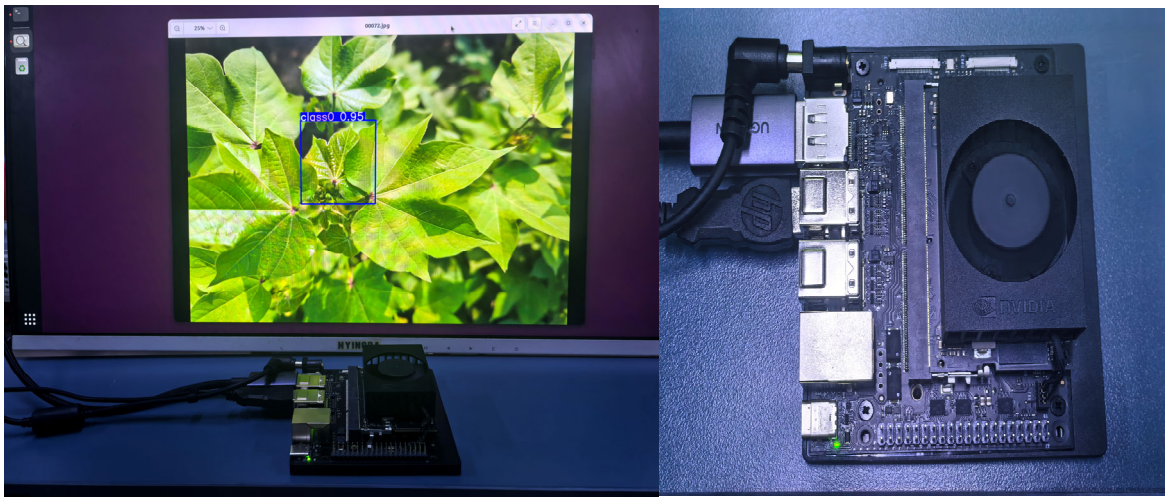
Although the proposed model demonstrated excellent overall generalization performance, we observed some failure cases under extreme conditions. When multiple wheat ears overlapped significantly, the model occasionally failed to detect underlying objects because of the lack of effective textural features. This failure explained why the recall rate was slightly lower than that of YOLOv8s. Furthermore, out-of-focus wheat ears located at the edges of the image, or those whose color was extremely similar to the dry background, were prone to generating false positives with low confidence scores.

3.6. Edge Deployment Experiment on Jetson Orin Nano

To verify the deployment feasibility of the proposed model in actual edge computing scenarios, this study deployed the trained model on an NVIDIA Jetson Orin Nano edge computing platform, with TensorRT FP16 precision for inference acceleration. The configuration of the Jetson Orin Nano device is presented in Table 10 and Figure 7.

Table 10. Edge deployment environment configuration.

Item	Configuration	Item	Configuration
Operating system	Ubuntu 22.04	TensorRT	10.7.0.23
JetPack	6.2	OpenCV	4.10.0
PyTorch	2.0.0	Inference precision	FP16
CUDA	12.6.85	Acceleration framework	TensorRT



(**a**) (**b**)

Figure 7. Schematic diagram of deploying the proposed model on a Jetson Orin Nano device: (**a**) real-time inference visualization on the Jetson Orin Nano platform; (**b**) Jetson Orin Nano edge device for embedded inference.

To evaluate the practical deployment benefits of the lightweight optimization strategies proposed in this paper, we conducted comparative performance experiments between RT-DETR-r18 and MC-RT-DETR at the same input resolution. The primary evaluation metrics included engine size after model conversion, FPS, and mean inference latency. The results of the comparative tests are shown in Table 11.

Table 11. Comparison of deployment performance on Jetson Orin Nano.

Model	Engine Size (MB)	FPS	Mean Latency (ms)
RT-DETR-r18	42.7	57.9	17.53
MC-RT-DETR	25.2	65.9	15.48

According to the data in Table 11, RT-DETR-r18 achieved an inference speed of 57.9 FPS on this edge device, with a model size of 42.7 MB. By contrast, the MC-RT-DETR realized an approximately 13.8% improvement in inference speed, reaching 65.9 FPS, while the average latency per frame decreased from 17.53 ms to 15.48 ms. More importantly, owing to its lightweight design, the MC-RT-DETR engine size was significantly reduced to 25.2 MB, representing a reduction of approximately 41.0% compared with that of the RT-DETR-r18 model.

In conclusion, MC-RT-DETR achieved more efficient inference speeds while reducing complexity and memory usage; its throughput of 65.9 FPS considerably exceeded the 30 FPS real-time standard typically required for field-based visual detection systems. Figure 7 shows the model's deployment on the device and its detection performance. The results of the deployment experiments demonstrated that the model achieved good balance between inference accuracy, speed, and compactness, making it highly suitable for deployment on edge devices in fields.

4. Discussion

4.1. Trade-Off Analysis Between Model Accuracy and Efficiency

Directly replacing the ResNet18 trunk network with MobileNetV3 significantly reduced the complexity of the model and led to a significant decrease in the mAP50-95 score. This finding highlights a key biological reality of the task: The recognition of cotton top buds is highly dependent on their robust fine-grained characteristic representation. The integrated C2f module successfully restored the baseline mAP50 indicator, indicating that the enhanced multiscale feature fusion can effectively compensate for the characteristic defects inherent in lightweight trunk networks. In addition, the introduction of the WIoU loss function significantly improved mAP50-95, showing that regression optimization for specific bounding boxes plays an important role in high-precision positioning of small agricultural targets.

4.2. Mechanism of WIoU in Top Bud Localization

Cotton top buds are usually located at the top of the stems and branches and are often wrapped in dense young leaves. Therefore, their visual exposure area is very small and extremely vulnerable to different degrees of obstruction. In actual testing, the prediction box may only cover part of the target area or contain considerable surrounding blade backgrounds. Therefore, improving the return quality of the bounding box is a key issue in cotton top bud detection. The WIoU loss function solves this challenge through dynamic nonmonotonous focusing. By dynamically adjusting the gradient gain of samples of different qualities, it forces the network to prioritize ordinary instances with valid and optimized positioning errors. Compared with the traditional GIoU, WIoU shows better high-precision bounding box fitting ability. However, the iterative experiment revealed an inherent trade-off: The initial implementation of WIoU led to a decline in the recall rate. This outcome showed that overly strict positioning penalties may cause the algorithm to be extremely conservative, making the network hesitate or misclassify severely obscured or blurred top buds as unrecognizable backgrounds. Through decoupled optimization strategy deployment, MixUp integration, Copy-Paste data enhancement, and appropriate loss function scaling, the proposed framework successfully increased the recall rate from 87.72% to 90.39%. This result verified that our decoupling method can effectively alleviate the network conservatism caused by strict regression constraints, thus taking into account the classification recall rate and spatial accuracy.

4.3. Application Prospects in Precision Cotton Topping

The dataset used in this study comes from cotton top bud detection research, and its original application background is related to chemical topping. However, from the perspective of visual perception, cotton top bud detection is not limited to chemical topping scenes. Whether it involves variable-rate chemical topping, mechanical topping, or robotic precision operations, detecting the position of top buds is a critical foundation. In variable-rate chemical topping, the proposed model can directly provide the precise spatial coordinates of the top buds to assist in the decision-making of spraying location to reduce pesticide waste and environmental impact. For automated mechanical topping, the proposed model can generate a reliable 2D detection algorithm.

5. Conclusions

This study proposes a lightweight visual perception framework based on an improved RT-DETR architecture, which is used for real-time edge detection of cotton top buds to achieve accurate topping. By replacing the ResNet18 backbone network with MobileNetV3 and integrating the C2f module, the framework reduces the computational complexity while maintaining feature characterization ability. In addition, we employ WIoU to optimize the accuracy of bounding box regression and combine it with MixUp, Copy-Paste, and adjusted weights for the bounding box loss to enhance the model's robustness. The experimental results show that the model achieves good balance between accuracy and efficiency: The proposed model realizes mAP50-95 of 83.59%, the number of parameters is greatly reduced to 11.01 M, and the computational load is decreased to 24.7 GFLOPs. More importantly, the deployment test on a Jetson Orin Nano edge device shows that the proposed model reaches 65.9 FPS in FP16 TensorRT reasoning mode, fully meeting the needs of real-time detection in fields.

Although the MC-RT-DETR model proposed in this paper demonstrates significant advantages in terms of deployment efficiency on edge devices, this study still has certain limitations. The core contribution of this improvement lies in significantly reducing model complexity to accommodate agricultural edge computing capabilities, rather than pushing the upper limit of absolute detection accuracy. Consequently, compared with the parameter-intensive RT-DETR-r18 baseline model, the proposed model exhibits a slight decline—within a controllable range—in absolute metrics such as recall. Moreover, the validation in this study primarily relies on a single publicly available cotton top bud dataset [30], which remains limited in terms of diversity regarding cotton cultivars, geographic distribution, phenological stages, and extreme variations in field illumination. At the algorithmic level, although the model demonstrates good robustness to local occlusions, a risk of false negatives in extremely dense scenarios with severe branch and leaf overlap remains. Furthermore, the current model's training relies on fully supervised, manually annotated data. To address these issues, future research will focus on in-depth expansion based on real-world operational scenarios using agricultural machinery platforms. First, we will construct a proprietary dataset covering various phenological stages, and variable weather conditions, and explore semi-supervised learning paradigms to re-duce the cost of manual annotation. Second, to address the challenge of extreme occlusion in static 2D images, we plan to leverage the continuous video stream characteristics of agricultural machinery operations by introducing a multiframe temporal feature fusion mechanism to compensate for missing single-frame information through a dynamic perspective. Finally, we intend to integrate the improved 2D object detection algorithm with RGB-D cameras or stereo vision systems to obtain 3D spatial coordinates of top buds, thereby providing a truly end-to-end, high-precision spatial positioning closed-loop system for mechanized topping operations.

Author Contributions: Conceptualization, Y.K. and G.A.; methodology, Y.K. and X.Q.; software, Y.K., X.L. and W.C.; validation, Y.K., X.L. and M.H.; formal analysis, X.Q.; investigation, Y.K.; resources, G.A.; writing—original draft preparation, Y.K.; writing—review and editing, G.A.; visualization, Y.K. and W.C.; project administration, G.A.; funding acquisition, G.A. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the Natural Science Foundation of Xinjiang Uygur Autonomous Region under Grant (No.: 2023D01C52); the Project of the State Key Laboratory for Novel Software Technology, Nanjing University (No.: KFKT2026B29); the Key Project of Yili Normal University under Grant (No.: 2023YSZD004); and the Science and Technology Program of Ili Kazakh Autonomous Prefecture under Grant (No.: YZ2022YD001).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The original data presented in this study are openly available in the following repository: https://www.kaggle.com/datasets/song0927/cotton-top-bud (accessed on 19 February 2026). The code is available from the corresponding author upon reasonable request.

Acknowledgments: We are especially grateful to the developers of the dataset used in this study. We also thank the school for the experimental facilities and equipment and our supervisor and group members for their help along the way.

Conflicts of Interest: The authors declare no conflicts of interest.

Appendix A

Table A1. Notation Table.

Symbol	Description
z	Global feature vector generated by global average pooling
s	Channel weight vector learned via fully connected layers
$\sigma(\cdot)$	Sigmoid activation function
B, Bgt	Predicted bounding box and ground-truth bounding box
C	The smallest enclosing convex box covering B and B^{gt}
$r(\beta)$	Dynamic nonmonotonic focusing factor in WIoU
β	Outlier degree of the predicted box
λreg	Bounding box regression loss function
λ	Interpolation coefficient following the beta distribution in MixUp
M	Binary mask corresponding to the pasted object area in Copy-Paste
$T(\cdot)$	Geometric transformation operation applied during Copy-Paste

Appendix B

Table A2. Per-Seed Test Set Results of the Multiseed Evaluation.

Model	Seed	P (%)	R (%)	mAP50 (%)	mAP50-95 (%)
RT-DETR-r18	0	95.24	92.58	96.39	84.01
RT-DETR-r18	42	94.08	92.53	95.47	81.61
RT-DETR-r18	123	96.67	91.81	96.02	80.30
RT-DETR-r18	3407	96.54	91.10	95.47	83.48
MC-RT-DETR	0	95.69	90.39	96.00	83.59
MC-RT-DETR	42	95.36	88.61	94.61	82.30
MC-RT-DETR	123	94.11	90.57	93.81	82.14
MC-RT-DETR	3407	96.19	89.80	95.94	82.96

References

1. Tlatlaa, J.S.; Tryphone, G.M.; Nassary, E.K. Unexplored Agronomic, Socioeconomic and Policy Domains for Sustainable Cotton Production on Small Landholdings: A Systematic Review. *Front. Agron.* **2023**, *5*, 1281043. [CrossRef]
2. Dai, J.; Tian, L.; Zhang, Y.; Zhang, D.; Xu, S.; Cui, Z.; Li, Z.; Li, W.; Zhan, L.; Li, C.; et al. Plant Topping Effects on Growth, Yield, and Earliness of Field-Grown Cotton as Mediated by Plant Density and Ecological Conditions. *Field Crops Res.* **2022**, *275*, 108337. [CrossRef]
3. Xu, Y.; Han, C.; Zhang, J.; Hu, B.; Ma, X.; Mao, H. Innovative Designs for Cotton Bionic Topping Manipulator. *Agriculture* **2024**, *14*, 1469. [CrossRef]
4. Cui, Y.; Li, P.; Xia, B. CTB-YOLO: A Lightweight Framework for Robust Cotton Terminal Bud Detection in Complex Field Environments. *Smart Agric. Technol.* **2026**, *13*, 101795. [CrossRef]
5. Zheng, Y. Object Detection and Recognition Based on Deep Learning. *Procedia Comput. Sci.* **2025**, *262*, 1337–1343. [CrossRef]
6. Shen, S.; Duan, F.; Tian, Z.; Han, C. A Novel Deep Learning Method for Detecting Strawberry Fruit. *Appl. Sci.* **2024**, *14*, 4213. [CrossRef]
7. Wu, J.; Zhang, J.; Tan, L.; Wu, Y.; Gao, K. Lightweight Transformer and Faster Convolution for Efficient Strawberry Detection. *Appl. Sci.* **2025**, *16*, 293. [CrossRef]
8. Bao, Y.; Qi, X.; Wang, H.; Liu, T.; Bai, Y. CE-Fusion Botanic: A Lightweight Leaf Disease Detection Model via Adaptive Local–Global Information Fusion. *Appl. Sci.* **2026**, *16*, 3177. [CrossRef]
9. Fan, X.; Sun, T.; Chai, X.; Zhou, J. YOLO-WDNet: A Lightweight and Accurate Model for Weeds Detection in Cotton Field. *Comput. Electron. Agric.* **2024**, *225*, 109317. [CrossRef]
10. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 1137–1149. [CrossRef] [PubMed]
11. Lin, T.-Y.; Dollár, P.; Girshick, R.; He, K.; Hariharan, B.; Belongie, S. Feature Pyramid Networks for Object Detection. In Proceedings of the 2017 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 2117–2125. [CrossRef]
12. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.-Y.; Berg, A.C. SSD: Single Shot MultiBox Detector. In *Proceedings of the 2016 European Conference on Computer Vision (ECCV), Amsterdam, The Netherlands, 11–14 October 2016*; Springer: Cham, Switzerland, 2016; pp. 21–37. [CrossRef]
13. Badgujar, C.M.; Poulose, A.; Gan, H. Agricultural Object Detection with You Only Look Once (YOLO) Algorithm: A Bibliometric and Systematic Literature Review. *Comput. Electron. Agric.* **2024**, *223*, 109090. [CrossRef]
14. Chen, K.; Zhu, L.; Song, P.; Tian, X.; Huang, C.; Nie, X.; Xiao, A.; He, L. Recognition of cotton terminal bud in field using improved Faster R-CNN by integrating dynamic mechanism. *Trans. Chin. Soc. Agric. Eng.* **2021**, *37*, 161–168. [CrossRef]
15. Song, P.; Chen, K.; Zhu, L.; Yang, M.; Ji, C.; Xiao, A.; Jia, H.; Zhang, J.; Yang, W. An Improved Cascade R-CNN and RGB-D Camera-Based Method for Dynamic Cotton Top Bud Recognition and Localization in the Field. *Comput. Electron. Agric.* **2022**, *202*, 107442. [CrossRef]
16. Peng, X.; Zhou, J.; Xu, Y.; Xi, G. Cotton top bud recognition method based on YOLOv5-CPP in complex environment. *Trans. Chin. Soc. Agric. Eng.* **2023**, *39*, 191–197. [CrossRef]
17. Zhao, L.; Peng, Q.; Lan, Y.; Kang, J.; Zhang, J.; Dai, J.; Chen, Y. CottonBud-YOLOv5s lightweight cotton bud detection algorithm. *Trans. Chin. Soc. Agric. Eng.* **2025**, *41*, 175–184. [CrossRef]
18. Zhao, L.; Kang, J.; Chen, Y.; Hu, B.; Chen, Y.; Huang, J.; Peng, Q. Real-Time Recognition and Dynamic Positioning Method for Cotton Terminal Buds Based on CottonBud-YOLOv5s Algorithm and RGBD Camera. *Smart Agric. Technol.* **2025**, *11*, 100975. [CrossRef]
19. Zhang, X.; Chen, L. Bud-YOLO: A Real-Time Accurate Detection Method of Cotton Top Buds in Cotton Fields. *Agriculture* **2024**, *14*, 1651. [CrossRef]
20. Fang, H.; Xu, Q.; Chen, X.; Zhang, Q.; Lu, J. Cotton top bud recognition method based on DFR-SN-YOLO. *Trans. Chin. Soc. Agric. Eng.* **2025**, *41*, 162–174. [CrossRef]
21. Xie, Y.; Chen, L. CBLN-YOLO: An Improved YOLO11n-Seg Network for Cotton Topping in Fields. *Agronomy* **2025**, *15*, 996. [CrossRef]
22. Carion, N.; Massa, F.; Synnaeve, G.; Usunier, N.; Kirillov, A.; Zagoruyko, S. End-to-End Object Detection with Transformers. In *Proceedings of the 2020 European Conference on Computer Vision (ECCV), Glasgow, UK, 23–28 August 2020*; Springer: Cham, Switzerland, 2020; pp. 213–229. [CrossRef]
23. Zhu, X.; Su, W.; Lu, L.; Li, B.; Wang, X.; Dai, J. Deformable DETR: Deformable Transformers for End-to-End Object Detection. In Proceedings of the 2021 International Conference on Learning Representations (ICLR), Virtual, 3–7 May 2021. [CrossRef]
24. Zhang, H.; Li, F.; Liu, S.; Zhang, L.; Su, H.; Zhu, J.; Ni, L.M.; Shum, H.-Y. DINO: DETR with Improved De-Noising Anchor Boxes for End-to-End Object Detection. In Proceedings of the 2023 International Conference on Learning Representations (ICLR), Kigali, Rwanda, 1–5 May 2023. [CrossRef]

25. Zhao, Y.; Lv, W.; Xu, S.; Wei, J.; Wang, G.; Dang, Q.; Liu, Y.; Chen, J. DETRs Beat YOLOs on Real-Time Object Detection. In Proceedings of the 2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 16–22 June 2024; pp. 16965–16974. [CrossRef]
26. Wang, S.; Jiang, H.; Yang, J.; Ma, X.; Chen, J.; Li, Z.; Tang, X. Lightweight Tomato Ripeness Detection Algorithm Based on the Improved RT-DETR. *Front. Plant Sci.* **2024**, *15*, 1415297. [CrossRef] [PubMed]
27. Zhang, Y.; Song, J.; Yu, X.; Ji, X. WMC-RTDETR: A Lightweight Tea Disease Detection Model. *Front. Plant Sci.* **2025**, *16*, 1574920. [CrossRef] [PubMed]
28. Guo, L.; Wang, X.; Zhang, Y.; Gong, J. RT-DETR-ME: A Lightweight Corn and Weed Detection Algorithm Based on Improved RT-DETR. *J. Biosyst. Eng.* **2025**, *50*, 410–423. [CrossRef]
29. Zhao, Z.; Chen, S.; Ge, Y.; Yang, P.; Wang, Y.; Song, Y. RT-DETR-Tomato: Tomato Target Detection Algorithm Based on Improved RT-DETR for Agricultural Safety Production. *Appl. Sci.* **2024**, *14*, 6287. [CrossRef]
30. Song, M.; Lan, Y.; Han, X.; Gou, X.; Wang, F.; Yang, W.; Cui, L. YOLOv8-CTB: A Lightweight and Real-Time Cotton Bud Detection under Field Conditions with Edge Deployment. *Clust. Comput.* **2026**, *29*, 121. [CrossRef]
31. Howard, A.; Sandler, M.; Chen, B.; Wang, W.; Chen, L.-C.; Tan, M.; Chu, G.; Vasudevan, V.; Zhu, Y.; Pang, R.; et al. Searching for MobileNetV3. In Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Republic of Korea, 27 October–2 November 2019. [CrossRef]
32. Sandler, M.; Howard, A.; Zhu, M.; Zhmoginov, A.; Chen, L.-C. MobileNetV2: Inverted Residuals and Linear Bottlenecks. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–23 June 2018; pp. 4510–4520. [CrossRef]
33. Jocher, G.; Qiu, J.; Chaurasia, A. Ultralytics YOLO 2023. Available online: https://github.com/ultralytics/ultralytics (accessed on 17 May 2026).
34. Rakesh, V.K.; Mazumdar, S.; Samanta, T.; Pandey, H.K.; Das, A.; Pal, S. Analysis of Hyperparameter Optimization Effects on Lightweight Deep Models for Real-Time Image Classification. *Sci. Rep.* **2026**, *16*, 17120. [CrossRef] [PubMed]
35. David, E.; Serouart, M.; Smith, D.; Madec, S.; Velumani, K.; Liu, S.; Wang, X.; Pinto, F.; Shafiee, S.; Tahir, I.S.A.; et al. Global Wheat Head Detection 2021: An Improved Dataset for Benchmarking Wheat Head Detection Methods. *Plant Phenomics* **2021**, *2021*, 9846158. [CrossRef] [PubMed]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.