agronomy

MDPI

Article

Real-Time Detection of Crops with Dense Planting Using Deep Learning at Seedling Stage

Shuolin Kong, Jian Li, Yuting Zhai, Zhiyuan Gao, Yang Zhou and Yanlei Xu *

College of Information and Technology, Jilin Agricultural University, Changchun 130118, China; 20201277@mails.jlau.edu.cn (S.K.); lijian@jlau.edu.cn (J.L.); 20221850@mails.jlau.edu.cn (Y.Z.); gzy@mails.jlau.edu.cn (Z.G.); zhouyang@jlau.edu.cn (Y.Z.)
* Correspondence: yanleixu@jlau.edu.cn

Abstract: Crop seedlings are similar in appearance to weeds, making crop detection extremely difficult. To solve the problem of detecting crop seedlings in complex field environments, a seedling dataset with four crops was constructed in this study. The single leaf labeling method was proposed as an alternative to conventional labeling approaches to improve the detection accuracy for dense planting crops. Second, a seedling detection network based on YOLOv5 and a transformer mechanism was proposed, and the effects of three features (query, key and value) in the transformer mechanism on the detection accuracy were explored in detail. Finally, the seedling detection network was optimized into a lightweight network. The experimental results show that application of the single leaf labeling method could improve the mAP0.5 of the model by 1.2% and effectively solve the problem of missed detection. By adding the transformer mechanism module, the mAP0.5 was improved by 1.5%, enhancing the detection capability of the model for dense and obscured targets. In the end, this study found that query features had the least impact on the transformer mechanism, and the optimized model improved the computation speed by 23 ms·frame^{-1} on the intelligent computing platform Jetson TX2, providing a theoretical basis and technical support for real-time seedling management.

Keywords: crop seedling detection; dense target detection; lightweight transformer; YOLOv5

Citation: Kong, S.; Li, J.; Zhai, Y.; Gao, Z.; Zhou, Y.; Xu, Y. Real-Time Detection of Crops with Dense Planting Using Deep Learning at Seedling Stage. *Agronomy* **2023**, *13*, 1503. https://doi.org/10.3390/agronomy13061503

Academic Editor: Roberto Marani

Received: 28 April 2023
Revised: 18 May 2023
Accepted: 23 May 2023
Published: 30 May 2023

1. Introduction

Precision agriculture, which is aimed at reducing the cost of agricultural production, environmental pollution and the automation of crop management [1], is gaining widespread attention and is being investigated by various agricultural researchers. In precision agriculture, the accurate acquisition of crop status and position is crucial for facilitating precise fertilization, weed control and full automation of crop management [2]. It serves as a key factor in reducing fertilizer waste, excessive herbicide use and controlling costs [3,4]. Crop detection becomes more challenging during the seedling stage, as crop seedlings bear a resemblance to weeds and are more susceptible to death from possible environmental factors. Hence, the design of a model capable of accurately detecting crop seedlings in complex environments becomes increasingly crucial.

In the past, crop detection often focused on individual crop species as targets and utilized conventional visual algorithms for detection [5,6]. Gai et al. [7] used the 2D connected components method, two-dimensional multiscale wavelet transformation and marker-controlled watershed segmentation algorithm to segment broccoli and lettuce. The accuracy for segmenting broccoli and lettuce achieved 92.4% and 96.6%, respectively. Chen et al. [8] extracted Gabor features from corn images and built a support vector machine (SVM) model to learn the Gabor features of corn images for corn detection. In addition, Hamuda et al. [9] used cauliflower HSV spatial images as input for cauliflower detection based on the dilation algorithm and moment method, which obtained a detection accuracy of 99.04%.

With the development of deep learning, convolutional neural networks (CNN) were developed, a vision algorithm with a stronger generalization ability and higher accuracy compared to traditional vision algorithms. CNN neural networks are widely used in precision agriculture and smart agriculture, such as automatic species identification [10], disease identification [11,12] and fruit ripeness analysis [13–15]. Most of the research on crop and weed localization is based on object detection networks, such as You Only Look Once (YOLO) [16–19] series models and Region-CNN (RCNN) series models [20–22]. Zou et al. [23] combined images with and without weeds to generate new weed images, and trained a semantic segmentation network called UNet, obtaining an accuracy of 92.21%. Punithavathi et al. [24] proposed a detection model based on Faster RCNN for crop and weed detection and used the extreme learning machine algorithm to optimize the hyperparameters of the deep learning model to obtain a higher detection accuracy. Chen et al. [25] detected weeds in sesame fields based on the YOLOv4 detection network and used local attention pooling to replace maximum pooling in spatial pyramid pooling and SEnet modules to replace logical modules in local attention pooling. The model obtained a 96.16% detection accuracy. Although these studies have provided a solid theoretical and experimental foundation for crop detection, the development of comprehensive automated crop management is still challenging. One of these challenges is that current studies mainly focus on crops that have grown to a degree where they can be easily distinguished from their surroundings. Another challenge is the precise localization of crops with dense planting (placing multiple seeds in a single planting hole), particularly during the seedling stage.

Additionally, we have identified several issues regarding the labeling process. In previous detection studies, researchers typically labeled the entire crop as the target, as shown in Figure 1a,b. However, this method presented three potential issues.

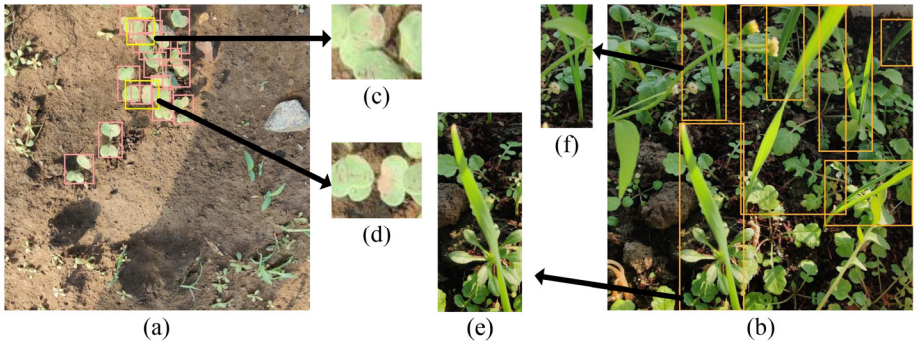


Figure 1. Schematic diagram of the whole crop labeling method. (**a**) Schematic of radish; (**b**) schematic diagram of wheat; (**c**) radish with only one leaf; (**d**) covered radish leaves; (**e**) wheat with only one leaf; (**f**) wheat with two leaves.

The first issue was that because of the dense nature of the crops, as shown in Figure 1d, the worker needs to be meticulous to identify multiple leaves belonging to the same crop. This process will produce errors in labeling. The second issue was that some crops can only be labeled to a single leaf due to occlusion, as shown in Figure 1c. As a result, there was a significant gap in the information about the same target features, making it difficult for the deep learning model to learn the feature patterns and reducing identification accuracy. Lastly, the third issue is that crops exhibit different growth rates, resulting in some crops having only one leaf, while others may have multiple leaves, as shown in Figure 1e,f. This inconsistency also reduced the detection accuracy. Although the whole crop labeling method can aid in identifying each crop, these three scenarios can lead to labeling errors. Additionally, in some cases, there may be a significant difference in the shape of the target, which can result in reduced accuracy for detecting the crop. Therefore, it is important to develop more effective labeling strategies to improve the accuracy of crop detection.

To address the issue of low accuracy in detecting crops from dense crop seedlings in complex environments, we constructed a crop seedling dataset that included crops such as

soybean, wheat, radish and cucumber, grown with different planting methods, sizes and growing environments. In addition, two labeling strategies were proposed in a scientific manner for dense planting crops. Furthermore, we proposed a deep learning structure transformer mechanism applied to computer vision that has been investigated recently. Since the transformer mechanism has been shown to be more resistant to interference and to extract features more rationally [26,27], it improves both the detection and recognition accuracy when combined with convolutional neural networks [28]. Therefore, we took inspiration from these studies that incorporated the transformer mechanism into their CNN-based model and expected to improve the detection accuracy of the model for various crop seedlings in complex environments.

In summary, we carried out the following work:

1. An image dataset was constructed, comprising seedlings from four different crops, all grown in environments with a substantial weed presence. Additionally, a dense planting method was used for some crops.
2. In order to improve the accuracy of the model for the detection of densely grown crops, two labeling strategies were proposed from the perspective of crop type.
3. A detection model for dense targets based on YOLOv5 and the transformer mechanism was proposed from the perspective of model structure.
4. Finally, in order to improve the detection efficiency, the model was lightened and improved based on the impact of three different features in the transformer mechanism on the accuracy.

The workflow diagram of the study is shown in Figure 2.

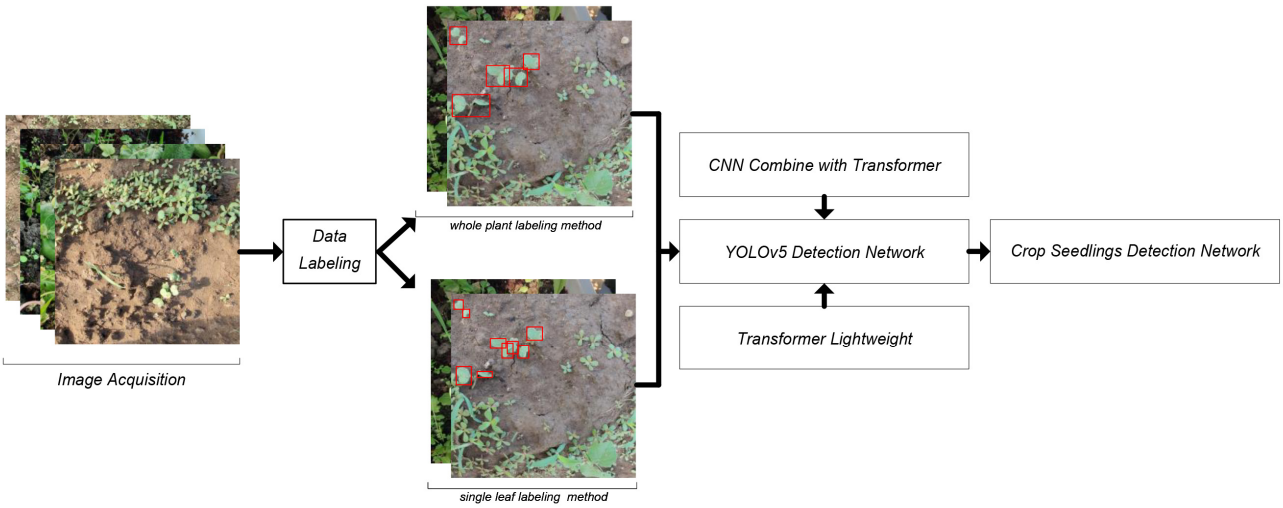


Figure 2. The research flowchart used in this study to detect crops at seedling stage.

2. Materials and Methods

2.1. Image Acquisition

The image capture device was a smartphone Oneplus8P (the manufacturer was BBK Electronics, Shenzhen, China), with a main camera lens of 48 million pixels, and the captured picture pixels were 3000 × 3000. The camera's ISO was set to 400, the color temperature was set to 5000 and the shutter speed was set to 1/50 s. The photographs were taken at a height ranging from 20 to 40 cm. In total, we collected 2140 digital RGB images in JPG format for use in this study.

In order to improve the model's generalization capacity and validate its efficacy in detecting a wide range of crops in complex environmental conditions, we meticulously built a dataset of crop seedlings with our own shooting. These data would be used to train and validate the model.

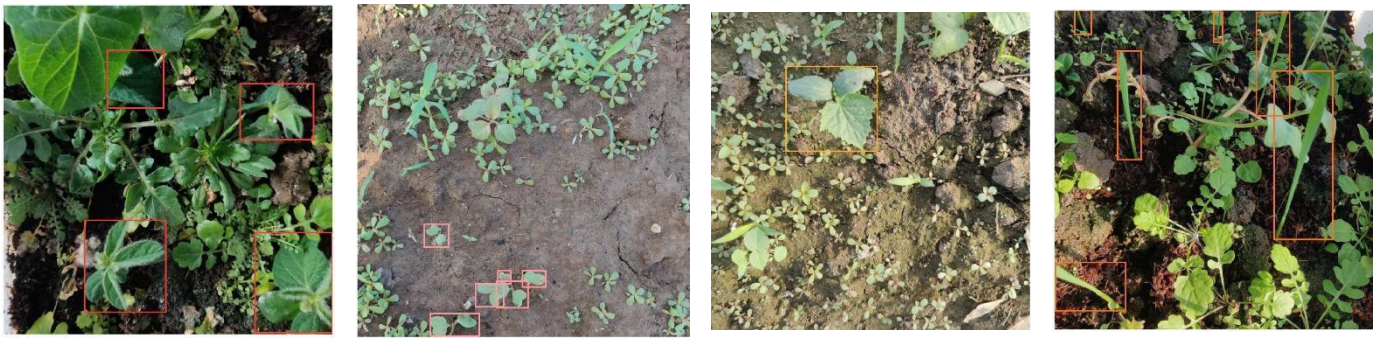
Four crops were targeted, namely soybeans, radishes, cucumbers and wheat. These crops were specifically chosen for two reasons. Firstly, soybean, radish, cucumber and

wheat are representative cash crops. Secondly, the seedling stages of various weed species, such as setaria viridis, eleusine indica, wild pea and petunia, bear resemblance to the seedling stages of the aforementioned crops.

The cultivation of these crops took place within a greenhouse located at Jilin Agricultural University in China. The greenhouse measures 5 m in width and 40 m in length. To create an environment rich in weed species, no weeding was conducted prior to crop planting and throughout the crop cultivation process. Specifically, soybeans and wheat were directly and randomly sown in the field, while cucumbers were planted at 20 cm intervals with 1–3 seeds placed in each planting hole. Radishes were also planted at 20 cm intervals, with more than 5 radish seeds placed in each planting hole. A total of 50 soybean and wheat seeds were planted, 27 cucumber seeds were planted and 30 planting holes were utilized for radishes.

During the seedling stage of the crops, the entire crop was photographed. Specifically, for soybeans, the seedling stage refers to the period from the emergence of cotyledons to the growth of the third true leaf. The same applies to cucumbers. For radishes, the seedling stage specifically refers to the period from the emergence of cotyledons to the growth of the first true leaf. As for wheat seedlings, the seedling stage refers to the period when the entire crop is less than 15 cm in height. Thirty specimens of each crop were planted and photographs were taken every three days starting from when the first pair of true leaves fully unfolded.

The crop seedling data comprised soybean, which was of medium size and planted sparsely, resulting in partial obscurity by weeds (Figure 3a), making detection slightly difficult (the size represents the proportion of the target crop in the entire image). Similarly, radish was small and densely planted, with seedlings not obscured by weeds but covered by other radish seedlings (Figure 3b), making it more challenging to detect. In addition, the cucumber was of medium size, planted sparsely and not shaded by other weeds, resulting in the least difficult detection (Figure 3c). Although wheat was planted sparsely, the seedlings were still be covered by other seedlings, which was the same for that of densely planted radishes (Figure 3d), which greatly increased the difficulty of detection. Detailed information on the data of each crop seedling is shown in Table 1. The dataset was split into training and test sets with the ratio of 0.8:0.2.



(**a**) (**b**) (**c**) (d)

Figure 3. Sample image of four crop seedling detection data. (**a**) Soybean; (**b**) radish; (**c**) cucumber; (**d**) wheat. Note: the color (HEX) of the detection box for soybean, radish, cucumber and wheat is FF3838, FF9D97, FF701F, FFB21D.

Table 1. Crop information and respective image collected in this study.

Type	Number	Size	Cover	Plant Method
Soybean	544	Medium	Partially	Sparse
Radish	515	Small	Mostly	Dense
Cucumber	503	Medium	None	Sparse
Wheat	578	Large	Mostly	Sparse

2.2. Labeling Strategy and Data Enhancement

To match the model input size and address the GPU memory limitations, we first reduced the image resolution from 3000 × 3000 to 640 × 640 using bilinear interpolation. Then, Labelimg 1.8.1 (an open source, efficient image labeling tool) was used to label the crop images and the tag data format was set to VOC format.

On the basis of the whole crop labeling method, we proposed the second labeling strategy as another possible option, referred to it as Strategy B, which utilized the single leaf labeling method, as shown in Figure 4. This labeling method has the following advantages. This method involved labeling a single leaf, thereby effectively reducing the labeling difficulty. Workers only needed to label each individual leaf without considering the logical relationship between the leaves. In addition, this method resulted in a more uniform target morphology and clearer features within the labeling box, particularly in cases of dense planting. By adopting this new labeling strategy, the study was able to overcome the limitations of the whole crop labeling method as stated in the introduction and improve the accuracy in crop detection.

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

Figure 4. Schematic diagram of the leaf labeling method. (**a**) Schematic diagram of radish; (**b**) schematic diagram of wheat.

To avoid model overfitting and to enhance the model performance, the amount of training data was expanded by employing various techniques. Two popular methods are image enhancement and mosaic enhancement. The image enhancement method involves applying random rotation and color dithering to the images. The color dithering operation adjusts various image properties such as image saturation, sharpness, brightness and contrast. The values of saturation and sharpness range from 0 to 3.1 and the values of luminance and contrast range from 1 to 2.1. The mosaic enhancement method takes a different approach by creating a single image from multiple randomly selected images. This process involved flipping and scaling the images, dithering the colors and finally stitching them together into a single image. The mosaic method allows the model to learn multiple images simultaneously, which improves the generalization ability of the model. A schematic diagram of the above two expansion methods is shown in Figure 5. The image enhancement was run online by the program, and it did not affect the original data. The enhancement probability we set was 50%, that is, there was a 50% possibility of enhancing the input image in one round of training.

(**a**) (**b**) (**c**) (**d**) (**e**)

Figure 5. Example of augmentations. (**a**) Original image of cucumber; (**b**) rotating image of cucumber; (**c**) original image of cucumber radish; (**d**) color dithering image of cucumber radish; (**e**) augmentative image with mosaic method.

2.3. Crop Seedlings Detection Network

The YOLOv5 is a single-stage target detection model, which is the most mature model in the YOLO series of models [29]. Compared to YOLOv3 [30] and YOLOv4 [31], YOLOv5 achieves a higher computational accuracy and minimal arithmetic power consumption. The model is continuously updated, making it more mature compared to YOLOv7 [32] and it outperforms its predecessors in various fields [33,34]. Therefore, YOLOv5 was used as the base detection network in this study.

The YOLOv5 structure consists of three main parts. Through these three structures, the model could effectively extract image features and locate key areas. The schematic diagram of the model structure is shown in Figure 6. The first part employed an improved cross-stage partial network (CSPNet) as the backbone network to extract the image-based feature information [35]. To improve the feature extraction, the faster feature extraction C3 module and the spatial pyramid pooling (SPP) that unifies multiple scale feature maps into the same size were added to CSPNet [20]. The second part was the neck network that extracted and processed the base image feature information output from the backbone network. To accomplish this, it used the feature pyramid network (FPN) structure [36], which consists of several neck blocks that perform computation and feature fusion. The neck blocks were responsible for aggregating the feature map information in the previous layer of neck blocks or the backbone network. The third component of the model was the detection head. Its primary purpose is for calculating relevant information of the detected targets in the detection box, such as the confidence level and the length and width adjustment value of the detection box. The detection head comprises three sub-components, each with a different feature map size, 76 × 76, 38 × 38, and 19 × 19, respectively. These sub-components detect small-sized, medium-sized and large-sized objects, respectively.

Transformer Neck Block

The transformer mechanism is a cutting-edge deep learning model with strong feature generalization and global feature extraction capabilities. The structure of the transformer mechanism is shown in Figure 7 and is comprised of several key components. Firstly, the Layer Norm represents the normalization process by layer, ensuring that the data are consistent and free from unwanted variations. Then, multi-head attention is the core of the transformer mechanism and is responsible for extracting global feature information. Lastly, the multilayer perceptron layer (MLP) module consists of two fully connected layers and an SiLU activation function, facilitating the transformation of the extracted features into a usable form.

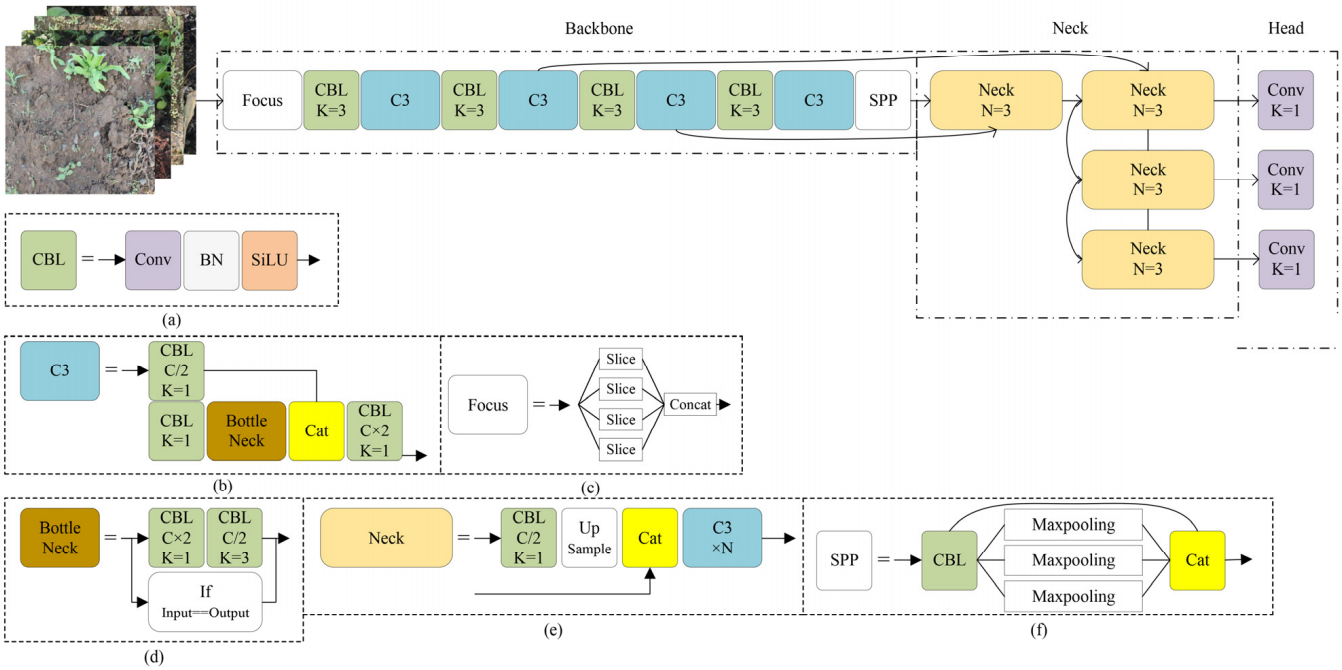


Figure 6. Diagram of the YOLOv5 detection network structure. Where, Conv is the convolutional layer, BN is the batch normalization layer and Cat is the concatenation operation. K is the convolutional kernel size, C is the number of channels and the step size of the convolutional kernel in the CBL module is 2. The second C3 block, the fourth C3 block and the SPP block are repeated 7, 1 and 3 times, respectively. (**a**) CBL module for downsampling operation; (**b**) C3 module is mainly responsible for extracting image features; (**c**) the Focus module is a downsampling module with no parameters; (**d**) BottleNeck is the module responsible for extracting features in the C3 module; (**e**) the Neck module is used to fuse features of different sizes; (**f**) SPP module can standardize the dimensions of feature drawings.

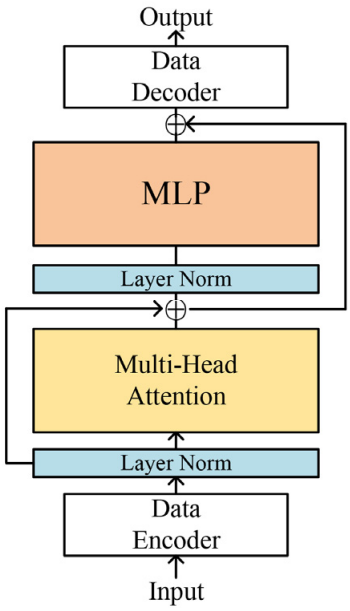


Figure 7. Diagram of the transformer mechanism structure in vision transformer model.

The transformer mechanism's abilities to extract global feature extraction relies on multi-head attention to compute the similarity between each image patch. The computation process of multi-head attention involves several steps. First, the input data pass through

the MLP to obtain three distinct feature sets, as shown in Equation (1), which are named Q, K and V, representing the query, key and value of the input image, respectively. Next, the similarity between Q and K is calculated as shown in Equation (2), where dk is the length of feature K and the Softmax function is used to obtain the weight value α of the similarity. Finally, the weight α is multiplied by V to output the features with global feature information, as shown in Equation (3).

$$Q, K, V = MLP(Input) \tag{1}$$

$$\alpha = Softmax(\frac{Q \cdot K}{\sqrt{d_K}}) \tag{2}$$

$$Output = \alpha \cdot V \tag{3}$$

This study aims to improve the information extraction capability of the model for dense targets. To achieve this, we incorporated the transformer mechanism into the YOLOv5 detection network because of its ability to extract global information. However, we found that adding the transformer mechanism directly to the backbone network resulted in high computational consumption and a weak inductive bias capability. To address this, we drew inspiration from previous studies that combined convolutional neural networks and the transformer mechanism [37,38]. Specifically, we added the transformer mechanism at the end of the neck of YOLOv5 to improve the detection capability of YOLOv5 for dense crop seedlings by using convolutional neural networks to bias the induction of features and reduce the number of feature channels. This approach helped to overcome the problem of weak bias induction and high computation of the transformer mechanism, while improving the detection efficiency. The improved neck block structure is shown in Figure 8.

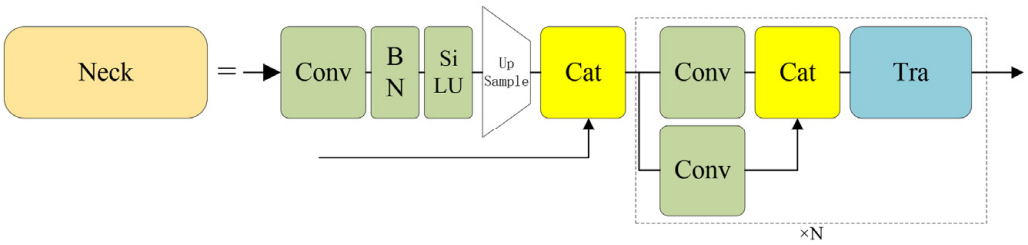


Figure 8. Diagram of the improved neck structure.

While the transformer mechanism has the potential to extract global feature information, which can enhance the model's ability to detect dense targets, it is prone to computational overload and has a tendency to overfit on small datasets [39]. In addition, it is worth noting that the transformer mechanism originated from the transformer model in the domain of natural language processing. In the case of text data, each piece of text has a distinct query, key and value due to the dense feature of the information contained in the text. Due to the sparse nature of the image information, it lacks the explicit query, key and value information that is necessary for the transformer mechanism to operate efficiently. As a result, Q, K and V for image data are simply features with varying information used by the transformer mechanism. However, the extraction of such information requires significant computational power, which is the primary reason for the slower processing speed of the transformer mechanism. In this study, we explored the impact of the three key features, i.e., Q, K and V, on the accuracy of the transformer mechanism. Due to the lack of clarity regarding the information carried by these features, we replaced them with the original input data. This approach allowed us to reduce the computational effort of the model while still examining the effect of these features on the accuracy. We tested two models that use a transformer mechanism, ViT [40] and Swim Transformer [41]. We investigated three types of replacement: replacement Q, replacement V and the simultaneous replacement of Q and V. Since the length of K is identical to that of Q, it is possible to substitute Q and K for each

other when calculating α. As a result, no experiments were conducted in this study for replacement K.

2.4. Training Settings and Hyperparameters

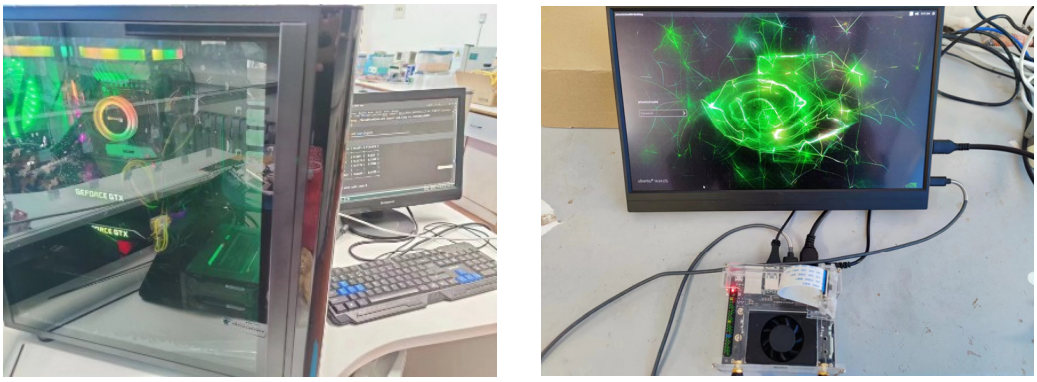
The optimization function used to train the model in this paper was the SGD function. The classification loss used the focal loss function to address the issue of imbalanced data caused by the higher number of crops in the densely grown crop images compared to the sparsely grown crops [42]. The detection loss was calculated using the IOU function [43], and the NMS (Non-Maximum Suppression) method [44] was used to filter redundant detection boxes. To effectively accelerate the model learning speed, we applied the OneCycleLR method for the learning rate. To obtain the optimal detection results of the model on the crop seedling dataset of this paper, we fine-tuned the hyperparameters of YOLOv5, as shown in Table 2. Among them, depth and width, which control the depth and width of the model, were kept constant. To obtain the optimal hyperparameters, the model underwent 500 training iterations, each consisting of 30 epochs. After each training iteration, we adjusted the hyperparameters and finally obtained the optimal set of hyperparameters with the highest accuracy during testing.

Table 2. Hyperparameters used in the training process.

Training Parameters	Values
depth	0.33
width	0.50
initial learning rate	0.009
final learning rate	0.071
momentum	0.95
optimizer weight decay	0.00045
focal loss alpha	0.5
focal loss γ	2

2.5. Experimental Equipment

The experiments were conducted on a Windows 10 operating system, using an NVIDIA Titan X GPU and Intel Xeon E5-2696 v3 CPU to train the deep learning models in PyTorch 1.7.0 with Python 3.6. The experimental device utilized for training the model is shown in Figure 9a.



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

Figure 9. Experimental device. (**a**) Experimental device utilized for training the model; (**b**) Jetson TX2.

The Jetson TX2 (as shown in Figure 9b) operating system was Ubuntu 18.04 using the deep learning framework Pytorch 1.8. The CPU was a CPU cluster consisting of a dual-core Denver2 processor and a quad-core ARM Cortex-A57, with 4 GB of LDDR4 memory and a 256-core Pascal GPU.

2.6. Performance Evaluation

To measure the performance of each trained model on the testing set, evaluation indicators were used in this study. Specifically, we evaluated the model performance by single species average precision (AP), mean detection precision (mAP), floating-point operations per second (FLOPs), computational speed of the training platform (server speed) and the computational speed of the carrying platform (TX2 speed). The unit of speed was ms·frame^{-1}, which was how long it took to calculate a frame. The AP was determined by precision and recall, the formula of precision is shown in Equation (4) and the formula of recall is shown in Equation (5), where *TP* is true positive, *FP* is false positive and *FN* is false negative. *TP*, *FP* and *FN* are determined by the IOU threshold. The IOU value is the overlap area between the detection box calculated by the model and the manually labeled detection box.

$$p = \frac{TP}{TP + FP} \tag{4}$$

$$r = \frac{TP}{TP + FN} \tag{5}$$

The formula for calculating AP is shown in Equation (6) and the value of the IOU was taken as 0.5 when evaluating the AP for a single crop. The mAP is the mean value of AP for the four species. The mAP had different IOU values. The mAP was named mAP0.5 when the value of the IOU was 0.5. The mAP0.5–0.95 was the mean value of the AP for the four species between IOU thresholds from 0.5 to 0.95 at intervals of 0.05 to calculate the AP. This index represented a more rigorous evaluation of the detection accuracy.

$$AP_i = \int_0^1 p_i(r_i)dr \tag{6}$$

Floating-point operations (FLOPs) measure the computational memory consumed by the model for each operation in the convolutional and fully connected layers during forward propagation of the model. The formula for calculating the FLOPs is displayed in Equation (7), where *Ci* is the convolutional layer input channel and *K* is the convolution kernel size. *H* and *W* are the height and weight of the convolutional layer output feature map, respectively, and *Co* is the output channel. *I* and O are the input and output numbers in the fully connected layers, respectively. The unit of *FLOPs* is *G* (Giga)

$$FLOPs = [\sum_{conv=1}^{n} (2C_I K^2 - 1)HWC_O] + [\sum_{full=1}^{n} (2I - 1)O] \tag{7}$$

3. Results

3.1. Results of Two Strategies and Transformer

The results obtained from the models on the testing set are shown in Table 3. In strategy A, the AP of the model with the transformer was improved for soybean (results of YOLOv5, YOLOv5ViT and YOLOv5ST: 90.0–91.4–91.9%), radish (77.3–80.7–81.1%) and cucumber (91.9–94.3–95.3%), demonstrating that the transformer mechanism could enhance the accuracy of the model for both dense and sparse sowing densities. However, we found that the AP for wheat was not significantly improved or even reduced. This is because wheat is planted densely and the wheat leaves are slender in shape, resulting in each detection box containing too many background features. As a result, the model was unable to focus on the region of interest of wheat.

This result also confirms our hypothesis that a large discrepancy in the feature information within the detection boxes of objects of the same category can lead to a reduction in precision. In strategy B, the same high AP was maintained for both sparsely planted cucumbers and soybeans, and an even higher AP was achieved for the model with the transformer mechanism. With strategy B, the detection box size of wheat was reduced, making each detection box contain only one leaf and reducing the feature gap in the detection box of the wheat. However, the characteristic of wheat leaves still caused too many

background features to be included in the box, resulting in more dense detection boxes and a lower wheat AP for YOLOv5. Nonetheless, we observed that the model with transformer mechanism significantly improved the detection accuracy of wheat relative to strategy A (results of strategy A: 73.2–73.7–72.9%; results of strategy B: 72.7–74.0–73.1%), indicating that the transformer mechanism is well-suited for detection in complex environments and medium-sized targets.

Table 3. The detection results of the whole crop labeling (strategy A) method and the single leaf labeling method (Strategy B) on the testing dataset.

Strategy	Model	mAP0.5 (%)	mAP0.5–0.95 (%)	AP (%)				FLOPs (G)	Server Speed (ms·frame^{-1})	TX2 Speed (ms·frame^{-1})
				Soybean	Radish	Cucumber	Wheat			
A	YOLOv5	83.1	52.8	90.0	77.3	91.9	73.2	4.7	13	106
	YOLOv5 ViT	85.0	53.8	91.4	80.7	94.3	73.7	5.3	17	145
	YOLOv5 ST	85.3	53.7	91.9	81.1	95.3	72.9	18.0	19	184
B	YOLOv5	84.3	52.6	91.4	81.9	91.3	72.7	4.7	13	106
	YOLOv5 ViT	85.8	53.3	90.9	83.9	94.2	74.0	5.3	17	145
	YOLOv5 ST	85.9	53.9	91.4	84.3	94.9	73.1	18.0	19	184

The training accuracy curves for YOLOv5, YOLOv5 ViT and YOLOv5 ST are shown in Figure 10. The curves showed that all three models achieved convergence after about 100 rounds. Because the underlying networks were based on YOLOv5, the curves displayed similar trends across the three models, as shown in Figure 10a. The model using strategy A had a large oscillation of the accuracy curve, especially before 100 epochs, as shown in Figure 10b. The large oscillation represents the difficulty of the model to learn the feature patterns, thereby highlighting the issue of a large gap in the information of the target individual features that existed in the overall labeling method. The model using strategy B, as shown in Figure 10c, had a lower initial accuracy due to the greater number of targets to be detected in each image. However, the gap in each individual feature's information was relatively small, the oscillation amplitude was lower and the accuracy improved more rapidly.

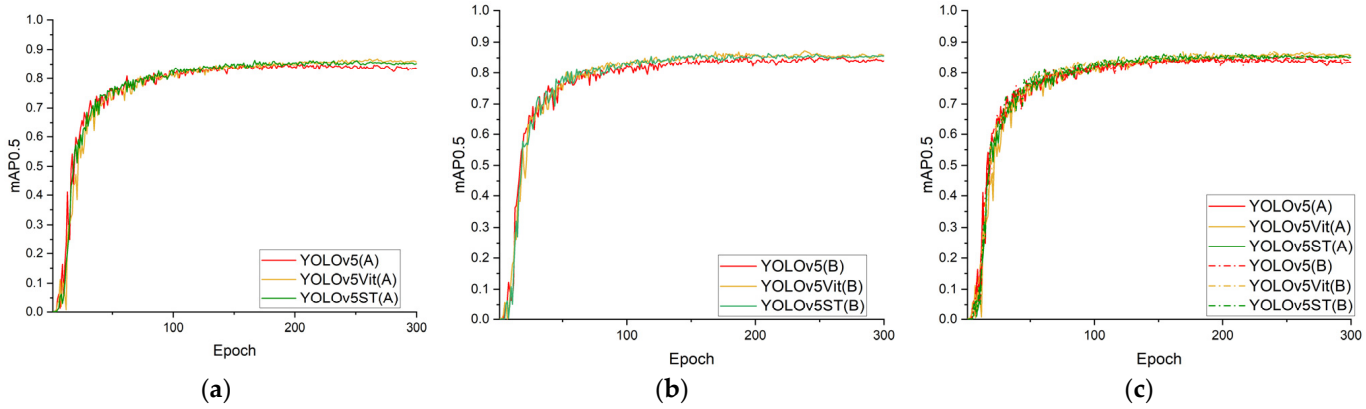


Figure 10. Training curves for the two strategies. (**a**) Comparison of strategy A; (**b**) comparison of strategy B; (**c**) comparison of the two strategies simultaneously.

Although the mAP of strategy B did not show a significant improvement compared to strategy A, it has practical advantages in detecting individual crops. As shown in the detection map of radishes in Figure 11, strategy B provided information on each crop, even if it was only partial, as shown in Figure 11e–g. In contrast, strategy A produced

missed individuals as shown in the detection images b, c and d. Therefore, despite the slight difference in the mAP, strategy B can provide more comprehensive and accurate information on individual crops, resolving the problem of difficult feature learning due to strategy A.

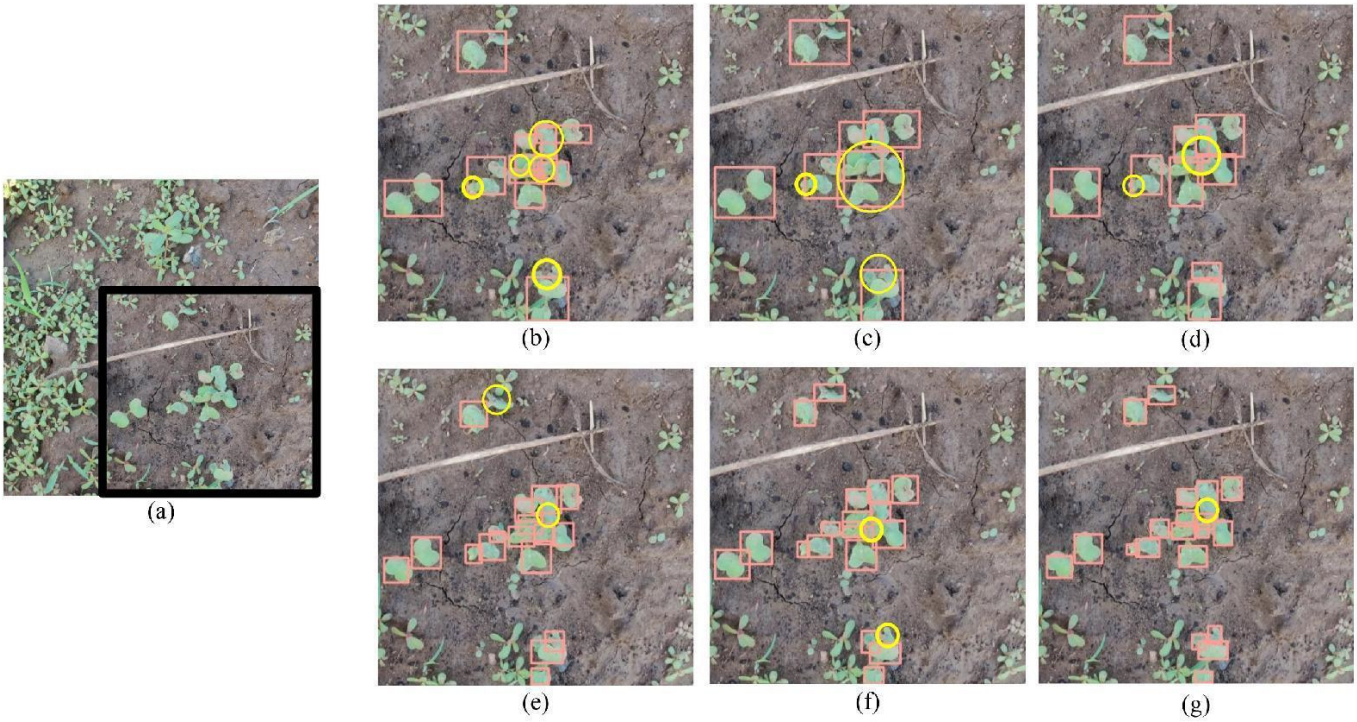


Figure 11. Examples of radish detection. (**a**) Original radish image; (**b**) detection image of YOLOv5 with strategy A; (**c**) detection image of YOLOv5ViT with strategy A; (**d**) detection image of YOLOv5ST with strategy A; (**e**) detection image of YOLOv5 with strategy B; (**f**) detection image of YOLOv5ViT with strategy B; (**g**) detection image of YOLOv5ST with strategy B. Note: yellow circled targets are missed.

The results of the wheat assay are shown in Figure 12. Although the whole crop labeling method was used (strategy A), multiple detection boxes for one crop were produced in the red circles (Figure 12b,c). This phenomenon is more akin to the single leaf labeling method. In addition, the phenomenon observed in Figure 12b highlighted the potential issue that crops with different numbers of leaves can lead to box selection confusion problems. The results in Figure 12c highlighted another potential issue of the presence of crop shading, as a result, the model may interpret two parts of the same crop as two crops. Furthermore, it was also verified that when the background within the detection box was too complex, it could interfere with the model's ability to effectively identify and classify the crop. Therefore, it was verified that the single leaf tagging method is more suitable for dense crop detection. These findings prove that the single leaf marker method may be more suitable for crop detection in high density fields.

In Figure 12e, the object detection results produced by the YOLOv5 model followed strategy B but did not frame the obscured parts (indicated by blue circles). While the detection model with the transformer successfully framed the two parts that led to a separation due to occlusion in one box, verifying that the transformer effectively provides global feature extraction capability.

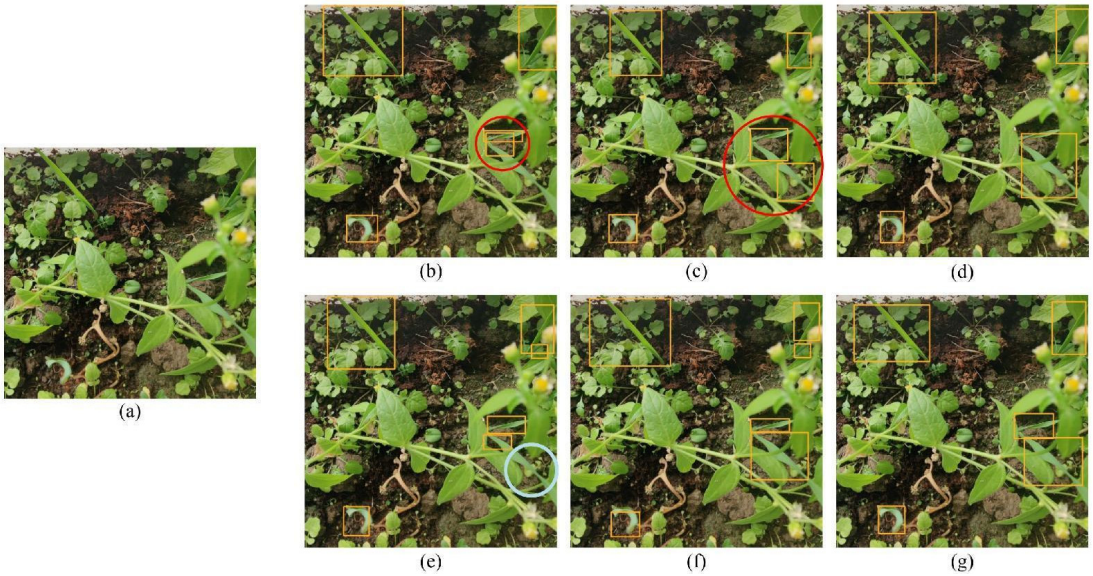


Figure 12. Examples of wheat detection. (**a**) Original wheat image; (**b**) detection image of YOLOv5 with strategy A; (**c**) detection image of YOLOv5ViT with strategy A; (**d**) detection image of YOLOv5ST with strategy A; (**e**) detection image of YOLOv5 with strategy B; (**f**) detection image of YOLOv5ViT with strategy B; (**g**) detection image of YOLOv5ST with strategy B. Note: red circled targets are not in compliance with the strategy. Blue circled targets are the missing parts.

For sparsely grown crops such as cucumbers and soybeans (Figure 13), even though the background was more complex, most of the models achieved a higher detection accuracy and produced satisfactory detection results even without any additional improvements. These findings suggested that there is no need to use strategy B for fields with low plant density during detection.

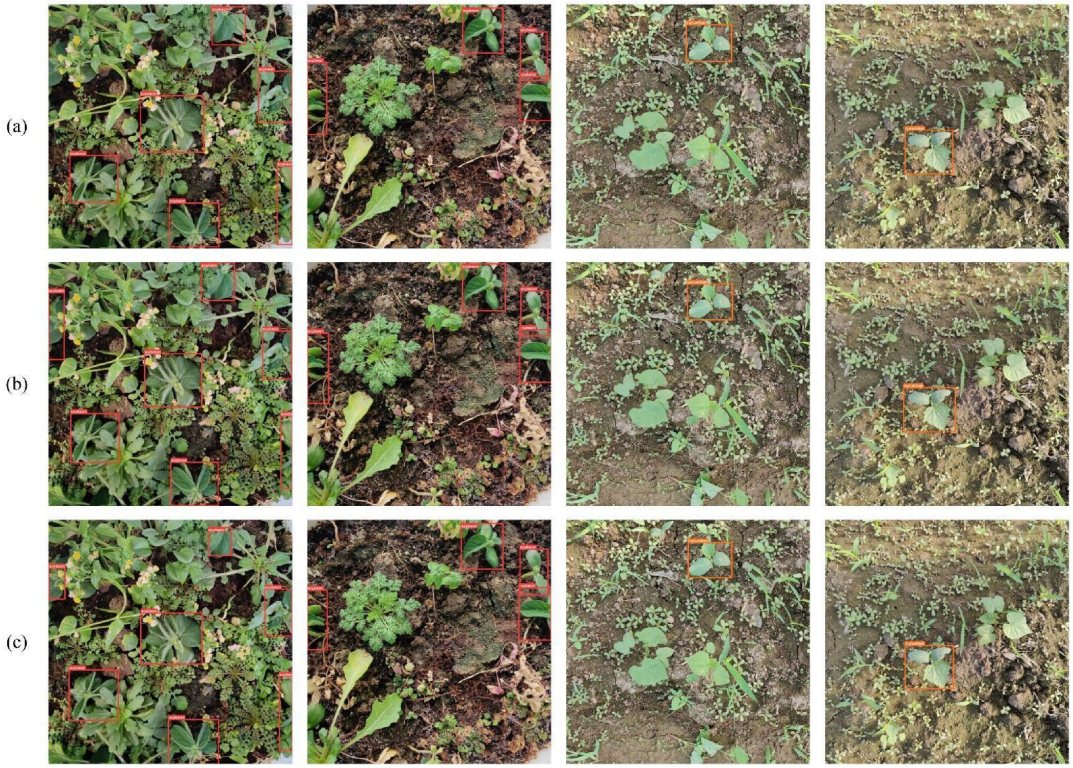


Figure 13. Examples of cucumber and soybean detection. (**a**) Detection image of YOLOv5; (**b**) detection image of YOLOv5ViT; (**c**) detection image of YOLOv5ST.

3.2. Results of the Different Transformer Detection Head

The results of eliminating the feature extraction module in the transformer mechanism are shown in Table 4.

Table 4. Detection results of the lightweight transformer on the testing dataset.

Strategy	Model	Eliminate Component	mAP0.5 (%)	mAP0.5–0.95 (%)	AP/%				FLOPs (G)	Server Speed (ms·frame^{-1})	TX2 Speed (ms·frame^{-1})
					Soybean	Radish	Cucumber	Wheat			
A	YOLOv5 ViT	-	85.0	53.8	91.4	80.7	94.3	73.7	5.3	17	145
		Q	84.2	51.9	90.9	79.9	93.2	72.6	5.0	16	128
		V	84.4	52.1	91.4	80.1	94.0	72.2	5.0	16	128
		QV	83.7	51.5	89.0	79.3	93.7	72.6	4.9	15	114
	YOLOv5 ST	-	85.3	53.7	91.9	81.1	95.3	72.9	10.7	19	184
		Q	84.4	52.3	90.7	81.1	94.7	71.1	9.4	18	164
		V	85.0	53.8	91.4	80.7	94.3	73.7	9.4	18	164
		QV	84.2	51.9	90.9	79.9	93.2	72.6	8.1	17	149
	YOLOv5	-	84.4	52.1	91.4	80.1	94.0	72.2	4.7	13	106
B	YOLOv5 ViT	-	85.8	53.3	90.9	83.9	94.2	74.0	5.3	17	145
		Q	84.6	52.9	91.2	79.9	93.7	73.7	4.9	16	128
		V	85.0	52.1	90.2	82.4	93.2	74.1	4.9	16	128
		QV	83.2	51.9	90.2	78.1	92.6	72.1	5.1	15	114
	YOLOv5 ST	-	85.9	53.9	91.4	84.3	94.9	73.1	10.7	19	184
		Q	83.8	52.1	89.7	79.2	93.5	72.8	9.4	18	164
		V	84.1	52.8	90.3	80.1	93.1	73.0	9.4	18	164
		QV	83.1	51.1	89.4	78.7	92.6	71.8	8.1	17	149
	YOLOv5	-	84.3	52.6	91.4	81.9	91.3	72.7	4.7	13	106

Benefiting from the window multi-head self-attention structure of the swim transformer, the mAP0.5 of YOLOv5ST was the highest among strategies A and B. Meanwhile, the elimination of feature blocks made the model faster. The elimination of V increased the computational speed to 164 ms·frame^{-1}, while the computational speed after eliminating Q and V increased to 149 ms·frame^{-1}. However, with the elimination of feature blocks, the mAP0.5 of YOLOv5ST decreased significantly. In strategy B, the computational accuracy of YOLOv5ST with feature V removed was reduced by 1.8%, and the simultaneous removal of Q and V led to a reduction in the computational accuracy of 2.8%.

The model robustness was higher for ViT with a simple structure. In strategy B, YOLOv5ViT reduced the mAP0.5 by 0.8% after eliminating V, which was smaller than that of YOLOv5ST. The simultaneous elimination of Q and V led to a reduction of 2.6% in the mAP0.5, which was similar to the reduction in YOLOv5ST. This demonstrated that the simultaneous removal of Q and V led to a reduction in the global feature extraction capability of the transformer. Although the improvement of the YOLOv5ViT computation speed was not as obvious as YOLOv5ST, which was only 23 ms·frame^{-1} faster on TX2, the YOLOv5ViT computation speed remained lower due to the relatively simple structure. The impact of removing the feature extraction module on the AP of cucumber was minimal. This was because the cucumber target was more prominent and the environment had fewer similar weeds without occlusions, indicating that YOLOv5 can detect this class of simple targets with a higher AP even without using the transformer mechanism.

In summary, YOLOv5ViT was the least sensitive to the effects of feature removal and had a faster speed, so YOLOv5ViT with V removed was used for further comparisons with other YOLO series models.

3.3. Comparison with Other YOLO Models

The results of each series of YOLO models are shown in Table 5. The YOLOv5 model used in this study had smaller FLOPs than the latest YOLOv7 model, which were slightly faster to compute in the server. Since there was no structural reparameterization in YOLOv5,

the computation speed in TX2 was faster than that in YOLOv7. This was the reason why we chose YOLOv5, a more mature model. The model with the transformer mechanism proposed in this paper had the same accuracy in the mAP0.5–0.95 as YOLOv7 in strategy A. Although the mAP0.5 of soybean and cucumber in strategy B was slightly lower than that of YOLOv7, both remained above 90%. The faster computation speed of our model on TX2 makes it more suitable for future embedding of the algorithm into small intelligent platforms for automated crop management applications.

Table 5. The detection results of different YOLO models on the testing dataset.

Strategy	Model	mAP0.5 (%)	mAP0.5–0.95 (%)	AP (%)				FLOPs (G)	Server Speed (ms·frame^{-1})	TX2 Speed (ms·frame^{-1})
				Soybean	Radish	Cucumber	Wheat			
A	YOLOv3	53.7	36.3	65.9	37.2	79	36.7	24.5	38	207
	YOLOv4	63.6	41.4	75.4	52.1	87.7	39.1	22.3	27	244
	YOLOv5	83.4	50.6	90	77.3	91.9	73.2	4.7	13	106
	YOLOX	83.2	49.9	90.6	79.1	91.8	71.3	15.2	22	121
	YOLOv7	83	52.1	90.4	74.6	92.1	72.9	26.7	15	141
	Ours	84.4	52.1	91.4	80.1	94	72.2	5.0	16	128
B	YOLOv3	53.9	35.5	64.7	38.4	78.9	33.5	24.5	38	207
	YOLOv4	63.3	40.7	75.7	52.6	88.7	36.3	22.3	27	244
	YOLOv5	84.3	52.6	91.4	81.9	91.3	72.7	4.7	13	106
	YOLOX	83.6	51	90.6	80.2	92.8	70.9	15.2	22	121
	YOLOv7	84.9	52.8	91.1	80.5	94.6	73.5	26.7	15	141
	Ours	85	52.1	90.2	82.4	93.2	74.1	4.9	16	128

4. Discussion

Accurate crop detection during the seedling stage enables the reduction in the damage inflicted by agricultural robots on crops, while simultaneously enhancing the efficiency of tasks such as fertilization and weed control. Research on crop detection using deep learning is quite extensive, such as Zou et al. [23] who employed advanced image generation techniques and captured crop images with complex backgrounds. However, their segmentation model required additional algorithmic processing to obtain the precise location of the crops. Furthermore, their computational speed was reported as 51 ms, whereas our proposed model achieved a faster computation speed of only 16 ms. Chen et al. [8] detected weeds in sesame fields based on the YOLOv4 detection network, but only one crop species was targeted. The same problem also occurred in the study by Hamuda [9] et al. The study conducted by Punithavathi et al. [24] utilized a dataset consisting of six crop species and eight weed species. However, the crops in this dataset were grown in an environment with a sparse weed presence and their computational speed was reported as 43 s, which was significantly slower compared to our model.

In our study, we also observed that the use of the single leaf labeling method resulted in a lower mAP0.5–0.95 accuracy compared to the whole crop labeling method. We speculate that this discrepancy may be attributed to the denser distribution of targets within the images when using the single leaf annotation method, which challenged the model's ability to achieve a higher precision detection. However, the model still demonstrated a good approximation of the target position, leading to improved accuracy in mAP0.5.

In the experiments eliminating Q and V, eliminating V had the lowest influence on the accuracy, while eliminating both Q and V had the highest influence on the accuracy, as shown in Table 4. The transformer computed the Q, K and V features of the input data, calculated the similarity weights between Q and K, and multiplied these similarity weights with V to obtain the global characteristics of V. Therefore, the results can show that the reason for the decrease in accuracy due to the simultaneous elimination of Q and V may have been that the transformer lost the ability to acquire global characteristics because the

similarity weights carried the information of V (at this time, both Q and V were equal to the input data) due to our elimination of Q when calculating the similarity weights between Q and K.

In addition, we noticed that some studies used deep learning techniques for the simultaneous detection of crops and weeds. However, this paper detected crops only. This decision was made because including weeds as additional targets in environments with dense weeds would result in imbalanced data for each class, leading to decreased model accuracy. Furthermore, labeling dense weeds is an impractical task. Moreover, weed control tasks do not require the use of deep learning or machine learning for weed detection. Once the position of crops is obtained, simple threshold segmentation methods can be used to detect weeds. Finally, by utilizing the coordinates of crops, avoidance strategies can be implemented to prevent accidental harm to crops during weed control operations.

This study contributes to the detection of dense crops and the lightweight optimization of the transformer, providing accurate positioning of crop seedlings before fertilization and spraying in robotic systems. And part of the crop seedling data and the actual detection video have been made public, which can be found at the following link 'https://github.com/xiaozhi101/crop_detection' (accessed on 27 April 2023). However, this study focused on the rectangular detection boxes, resulting in an insignificant improvement in wheat detection. Therefore, detection algorithms that can frame polygonal detection boxes will be investigated in the future to accurately detect crops with elongated leaves.

5. Conclusions

In this study, a target detection network was proposed for the efficient localization of crop seedlings in complex environments. The target detection network consisted of a YOLOv5 network and transformer module. First, to improve the accuracy and efficiency of the model for dense seedling detection, two labeling strategies, the whole crop labeling method (strategy A) and the single leaf labeling method (strategy B), were proposed. The results showed that the mAP0.5 could be improved from 83.1% to 84.3% using the whole crop labeling method, and from 77.3% to 81.9% for radishes (dense target). Second, the addition of the transformer module improved the mAP0.5 from 83.1% to 85% in strategy A and from 84.3% to 85.8% in strategy B, which effectively improved the detection precision of the model for complex environments and dense targets. Finally, the process of lightweight improvement of the transformer module revealed that feature extraction module V had the least impact on the features. By eliminating V, the computation speed was reduced by 1 ms·frame^{-1} in the server and 17 in the minicomputer TX2, and the mAP0.5 was reduced by only 0.6%, offering the possibility of real-time management of crop seedlings. Such results are particularly beneficial for complex environments and dense targets, highlighting the effectiveness of the transformer module in enhancing the performance of the model.

Author Contributions: Conceptualization, Y.X. and S.K.; Methodology, S.K. and J.L.; Validation, S.K. and Y.X.; Writing—Original Draft Preparation, Y.Z. (Yuting Zhai); Writing—Review & Editing, Y.Z. (Yang Zhou) and Y.X.; Supervision, Z.G. and Y.X.; Funding Acquisition, Y.X. All authors have read and agreed to the published version of the manuscript.

Funding: This study was financially supported by the Jilin Provincial Science and Technology Development Plan Project (Grant No. 20230202035NC), and the Science and Technology Development Plan Project of Changchun (Grant No. 21ZGN28).

Data Availability Statement: Not applicable.

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

References

1. Patricio, D.I.; Rieder, R. Computer vision and artificial intelligence in precision agriculture for grain crops: A systematic review. *Comput. Electron. Agric.* **2018**, *153*, 69–81. [CrossRef]
2. Jha, K.; Doshi, A.; Patel, P.; Shah, M. A comprehensive review on automation in agriculture using artificial intelligence. *Artif. Intell. Agric.* **2019**, *2*, 1–12. [CrossRef]

3. Fabregas, R.; Kremer, M.; Schilbach, F. Realizing the potential of digital development: The case of agricultural advice. *Science* **2019**, *366*, eaay3038. [CrossRef]
4. Tudi, M.; Ruan, H.D.; Wang, L.; Lyu, J.; Sadler, R.; Connell, D.; Chu, C.; Phung, D.T. Agriculture Development, Pesticide Application and Its Impact on the Environment. *Int. J. Environ. Res. Public Health* **2021**, *18*, 1112. [CrossRef] [PubMed]
5. Cubero, S.; Aleixos, N.; Molto, E.; Gomez-Sanchis, J.; Blasco, J. Advances in Machine Vision Applications for Automatic Inspection and Quality Evaluation of Fruits and Vegetables. *Food Bioprocess Technol.* **2011**, *4*, 487–504. [CrossRef]
6. Burgos-Artizzu, X.P.; Ribeiro, A.; Guijarro, M.; Pajares, G. Real-time image processing for crop/weed discrimination in maize fields. *Comput. Electron. Agric.* **2011**, *75*, 337–346. [CrossRef]
7. Gai, J.Y.; Tang, L.; Steward, B.L. Automated crop plant detection based on the fusion of color and depth images for robotic weed control. *J. Field Robot.* **2020**, *37*, 35–52. [CrossRef]
8. Chen, Y.J.; Wu, Z.N.; Zhao, B.; Fan, C.X.; Shi, S.W. Weed and Corn Seedling Detection in Field Based on Multi Feature Fusion and Support Vector Machine. *Sensors* **2021**, *21*, 212. [CrossRef]
9. Hamuda, E.; Mc Ginley, B.; Glavin, M.; Jones, E. Automatic crop detection under field conditions using the HSV colour space and morphological operations. *Comput. Electron. Agric.* **2017**, *133*, 97–107. [CrossRef]
10. Garibaldi-Marquez, F.; Flores, G.; Mercado-Ravell, D.A.; Ramirez-Pedraza, A.; Valentin-Coronado, L.M. Weed Classification from Natural Corn Field-Multi-Plant Images Based on Shallow and Deep Learning. *Sensors* **2022**, *22*, 3021. [CrossRef]
11. Thakur, P.S.; Sheorey, T.; Ojha, A. VGG-ICNN: A Lightweight CNN model for crop disease identification. *Multimed. Tools Appl.* **2023**, *82*, 497–520. [CrossRef]
12. Lee, S.H.; Goeau, H.; Bonnet, P.; Joly, A. New perspectives on plant disease characterization based on deep learning. *Comput. Electron. Agric.* **2020**, *170*, 12. [CrossRef]
13. Mu, Y.; Chen, T.S.; Ninomiya, S.; Guo, W. Intact Detection of Highly Occluded Immature Tomatoes on Plants Using Deep Learning Techniques. *Sensors* **2020**, *20*, 2984. [CrossRef]
14. Li, X.C.; Cai, C.L.; Zheng, H.; Zhu, H.F. Recognizing strawberry appearance quality using different combinations of deep feature and classifiers. *J. Food Process Eng.* **2022**, *45*, 11. [CrossRef]
15. Miragaia, R.; Chavez, F.; Diaz, J.; Vivas, A.; Prieto, M.H.; Monino, M.J. Plum Ripeness Analysis in Real Environments Using Deep Learning with Convolutional Neural Networks. *Agronomy* **2021**, *11*, 2353. [CrossRef]
16. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You only look once: Unified, real-time object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 779–788.
17. Ge, Z.; Liu, S.; Wang, F.; Li, Z.; Sun, J. Yolox: Exceeding yolo series in 2021. *arXiv* **2021**, arXiv:210708430.
18. Gao, A.; Geng, A.; Zhang, Z.; Zhang, J.; Hu, X.; Li, K. Dynamic detection method for falling ears of maize harvester based on improved YOLO-V4. *Int. J. Agric. Biol. Eng.* **2022**, *15*, 22–32. [CrossRef]
19. Hu, H.; Kaizu, Y.; Zhang, H.; Xu, Y.; Imou, K.; Li, M.; Huang, J.; Dai, S. Recognition and localization of strawberries from 3D binocular cameras for a strawberry picking robot using coupled YOLO/Mask R-CNN. *Int. J. Agric. Biol. Eng.* **2022**, *15*, 175–179. [CrossRef]
20. He, K.M.; Zhang, X.Y.; Ren, S.Q.; Sun, J. Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2015**, *37*, 1904–1916. [CrossRef]
21. Cheng, B.; Wei, Y.; Shi, H.; Feris, R.; Xiong, J.; Huang, T. Revisiting rcnn: On awakening the classification power of faster rcnn. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 453–468.
22. Rehman, Z.U.; Khan, M.A.; Ahmed, F.; Damasevicius, R.; Naqvi, S.R.; Nisar, W.; Javed, K. Recognizing apple leaf diseases using a novel parallel real-time processing framework based on MASK RCNN and transfer learning: An application for smart agriculture. *IET Image Process.* **2021**, *15*, 2157–2168. [CrossRef]
23. Zou, K.L.; Chen, X.; Wang, Y.L.; Zhang, C.L.; Zhang, F. A modified U-Net with a specific data argumentation method for semantic segmentation of weed images in the field. *Comput. Electron. Agric.* **2021**, *187*, 9. [CrossRef]
24. Punithavathi, R.; Rani, A.D.C.; Sughashini, K.R.; Kurangi, C.; Nirmala, M.; Ahmed, H.F.T.; Balamurugan, S.P. Computer Vision and Deep Learning-enabled Weed Detection Model for Precision Agriculture. *Comput. Syst. Sci. Eng.* **2023**, *44*, 2759–2774. [CrossRef]
25. Chen, J.Q.; Wang, H.B.; Zhang, H.D.; Luo, T.; Wei, D.P.; Long, T.; Wang, Z.K. Weed detection in sesame fields using a YOLO model with an enhanced attention mechanism and feature fusion. *Comput. Electron. Agric.* **2022**, *202*, 12. [CrossRef]
26. Shao, R.; Shi, Z.; Yi, J.; Chen, P.-Y.; Hsieh, C.-J. On the adversarial robustness of vision transformers. *arXiv* **2021**, arXiv:2103.15670.
27. Tuli, S.; Dasgupta, I.; Grant, E.; Griffiths, T.L. Are convolutional neural networks or transformers more like human vision? *arXiv* **2021**, arXiv:2105.07197.
28. Srinivas, A.; Lin, T.-Y.; Parmar, N.; Shlens, J.; Abbeel, P.; Vaswani, A. Bottleneck transformers for visual recognition. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 16519–16529.
29. You Only Look Once V5. Available online: https://github.com/ultralytics/YOLOv5 (accessed on 11 May 2022).
30. Redmon, J.; Farhadi, A. Yolov3: An incremental improvement. *arXiv* **2018**, arXiv:1804.02767.
31. Bochkovskiy, A.; Wang, C.-Y.; Liao, H.-Y.-M. Yolov4: Optimal speed and accuracy of object detection. *arXiv* **2020**, arXiv:2004.10934.
32. Wang, C.-Y.; Bochkovskiy, A.; Liao, H.Y.M. YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. *arXiv* **2022**, arXiv:2207.02696.

33. Yan, B.; Fan, P.; Lei, X.Y.; Liu, Z.J.; Yang, F.Z. A Real-Time Apple Targets Detection Method for Picking Robot Based on Improved YOLOv5. *Remote Sens.* **2021**, *13*, 1619. [CrossRef]
34. Yao, J.; Qi, J.M.; Zhang, J.; Shao, H.M.; Yang, J.; Li, X. A Real-Time Detection Algorithm for Kiwifruit Defects Based on YOLOv5. *Electronics* **2021**, *10*, 1711. [CrossRef]
35. Wang, C.-Y.; Liao, H.-Y.M.; Wu, Y.-H.; Chen, P.-Y.; Hsieh, J.-W.; Yeh, I.-H. CSPNet: A new backbone that can enhance learning capability of CNN. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, Seattle, WA, USA, 14–19 June 2020; pp. 390–391.
36. Lin, T.-Y.; Dollár, P.; Girshick, R.; He, K.; Hariharan, B.; Belongie, S. Feature pyramid networks for object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 2117–2125.
37. Petit, O.; Thome, N.; Rambour, C.; Themyr, L.; Collins, T.; Soler, L. U-net transformer: Self and cross attention for medical image segmentation. In Proceedings of the Machine Learning in Medical Imaging: 12th International Workshop, MLMI 2021, Held in Conjunction with MICCAI 2021, Strasbourg, France, 27 September 2021; pp. 267–276.
38. Li, Y.; Yao, T.; Pan, Y.; Mei, T. Contextual transformer networks for visual recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2022**, *45*, 1489–1500. [CrossRef] [PubMed]
39. Khan, S.; Naseer, M.; Hayat, M.; Zamir, S.W.; Khan, F.S.; Shah, M. Transformers in vision: A survey. *ACM Comput. Surv. (CSUR)* **2022**, *54*, 1–41. [CrossRef]
40. Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S. An image is worth 16 × 16 words: Transformers for image recognition at scale. *arXiv* **2020**, arXiv:2010.11929.
41. Liu, Z.; Lin, Y.; Cao, Y.; Hu, H.; Wei, Y.; Zhang, Z.; Lin, S.; Guo, B. Swin transformer: Hierarchical vision transformer using shifted windows. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, QC, Canada, 10–17 October 2021; pp. 10012–10022.
42. Lin, T.-Y.; Goyal, P.; Girshick, R.; He, K.; Dollár, P. Focal loss for dense object detection. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 2980–2988.
43. Rezatofighi, H.; Tsoi, N.; Gwak, J.Y.; Sadeghian, A.; Reid, I.; Savarese, S. Computer vision and pattern recognition. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 658–666.
44. Neubeck, A.; Van Gool, L. Efficient non-maximum suppression. In Proceedings of the 18th International Conference on Pattern Recognition (ICPR'06), Hong Kong, China, 20–24 August 2006; pp. 850–855.