agriculture

MDPI

Article

Analysis of the Impact of Different Improvement Methods Based on YOLOV8 for Weed Detection

Cuncai He, Fangxin Wan, Guojun Ma, Xiaobin Mou, Kaikai Zhang, Xiangfeng Wu and Xiaopeng Huang *

College of Mechanical and Electronical Engineering, Gansu Agricultural University, Lanzhou 730070, China; hecunc@st.gsau.edu.cn (C.H.)

* Correspondence: huangxp@gsau.edu.cn; Tel.: +86-138-9315-9327

Abstract: In response to the issues of missed detection, false positives, and low recognition rates for specific weed species during weed detection, a YOLOv8-based improved weed detection algorithm named EDS-YOLOv8 is proposed. Improvements were made in three main aspects. First, the YOLOv8 backbone network was enhanced with EfficientViT and RepViT architectures to improve the detection capability of dense-type weeds. Second, different attention mechanisms were added, such as SimAM and EMA, to learn 3D weights and achieve full fusion of features. BiFormer was introduced for dynamic sparse attention and resource allocation. Third, significant module improvement involved introducing dynamic snake convolution into the C2f module to further enhance detection capabilities for deformable objects, especially needle-shaped weeds. The improved model is validated on the established weed dataset. The results show that combining the original backbone network with dynamic snake convolutions yields the highest performance improvement. Precision, recall, mAP (0.5), and mAP (0.5:0.95) are improved by 5.6%, 5.8%, 6.4%, and 1%, respectively, and ablation experiments on the effects of the three improvement methods on model performance show that using EfficientViT as the backbone network while simultaneously improving the crucial module and adding the SimAM attention mechanism effectively enhances the model's performance. Precision, recall, mAP (0.5), and mAP (0.5:0.95) are improved by 6%, 5.9%, 6.4%, and 0.7%, respectively.

Keywords: attention mechanism; weed detection; YOLOv8

Citation: He, C.; Wan, F.; Ma, G.; Mou, X.; Zhang, K.; Wu, X.; Huang, X. Analysis of the Impact of Different Improvement Methods Based on YOLOV8 for Weed Detection. *Agriculture* **2024**, *14*, 674. https://doi.org/10.3390/agriculture14050674

Academic Editors: Maciej Zaborowicz and Jakub Frankowski

Received: 5 March 2024
Revised: 19 April 2024
Accepted: 21 April 2024
Published: 26 April 2024

1. Introduction

In 2015, the R-CNN [1] algorithm was proposed. Target detection algorithms ushered in a revolutionary development, from traditional target detection into the stage of deep learning. A series of algorithms such as AlexNet [2], VGG [3], GoogLeNet [4], and ResNet [5] have been proposed. In 2015, R. Girshick et al. proposed the Fast R-CNN [1] algorithm, which uses a method such as SS (selective search) to generate candidate boxes and then maps candidate regions of different sizes into a fixed-size feature map via RoI (Region of Interest). Ren et al. proposed Faster R-CNN [6], which introduces a RPN (Region Proposal Network) to realize target localization and detection. In 2016, Redmon et al. proposed the YOLOv1 [7] algorithm, and the target detection algorithm moved from a two-stage to a single-stage development direction. In 2017, Redmon et al. proposed YOLOv2 [8], and in 2018, Redmon et al. proposed YOLOv3 [9]. Since then, the YOLO series of algorithms has gone through the iterations of YOLOv4 [10],YOLOv5 [11],YOLOv6 [12], YOLOv7 [13], YOLOv8 [14], and the latest version, YOLOv9 [15].

Weeds represent one of the most significant threats to crop growth, acting as a major factor limiting crop yield. They interfere with the growth process in several ways: 1. Weeds compete with crops for essential resources such as nutrients and light, diminishing the available supply for crop plants [16]. 2. Moreover, weeds can serve as carriers for parasites and pathogens, facilitating the spread of pests and diseases within farmland. The extensive growth of weeds can occupy valuable space meant for crop growth, impeding the water

absorption of crops during irrigation. This, in turn, leads to reduced crop production due to water scarcity. 3. Certain weed species produce toxic seeds that pose a risk to human and livestock health if ingested accidentally, thereby threatening human safety. Given these challenges, weed control is an indispensable step in agricultural production. Currently, mainstream weed control methods include artificial weeding, which is labor-intensive and suitable primarily for small-scale fields; chemical weeding, which involves the indiscriminate spraying of herbicides, enabling its use in large areas but causing environmental, soil, and water pollution [17]; mechanized weeding, which offers high efficiency and is well suited for large-scale operations; and biological weeding, which utilizes fungi and bacteria to suppress weed growth, presenting an environmentally friendly option [18]. However, it can lead to the development of resistance, and its efficacy may vary. Each method has its advantages and limitations, and the choice of weed control approach depends on various factors such as crop type, field size, and environmental considerations.

Many scholars have achieved a lot of results in the research of deep-learning-based weed detection methods. Yuan Tao et al. [19] proposed a YOLOv4-based weed detection algorithm for paddy fields, which uses an inverse residual network to reduce the number of parameters, k-mean [20] to aggregate the a priori frames, and GAN to add noise to match the complex and changing backgrounds of the real situation. Yingli Cao et al. [21] proposed a weed recognition method based on DeepLabv3+ for rice field, which improves the ASPP module to realize the segmentation of weeds and rice in complex backgrounds. Wenqing Shang et al. [22] added double-threshold non-great suppression to Faster-RCNN to achieve 94.82% recall.

Although the above methods have improved weed detection performance to some extent, weed leaves exhibit diverse and intricate shapes, including scales, strips, thorns, and needles; moreover, natural lighting conditions vary significantly between day and night, and backgrounds are often complex and dynamic. Many weeds closely resemble crop plants during their growth stages, making it difficult for weed recognition models to accurately distinguish between them. This challenge impedes feature extraction for the weed recognition model, resulting in low detection accuracy and poor real-time performance in such scenarios. Therefore, there is still considerable room for improvement in weed detection methods based on deep-learning techniques.

This paper focuses on three common weed species as the research subject. To address issues such as missed detections, false positives in natural environments, as well as low accuracy in specific species such as needle-typed weeds, we propose the EDS-YOLOv8 model as an enhancement to the YOLOv8 target detection model. The main research contents and innovations are as follows:

(1) The backbone network is enhanced by replacing the original backbone network with two alternatives: RepViT, a lightweight network, and EfficientViT, a dense object detection network.
(2) In the improvement to the neck part, various attention mechanisms such as EMA, SimAM, and BiFormer are integrated to evaluate their impact on weed recognition accuracy. The optimal attention mechanism for the weed detection task is then selected based on comparative analysis.
(3) Partial enhancements are made to the C2f module and BottleNeck by substituting ordinary convolution with dynamic snake convolution. This dynamic snake convolution improves the capability to extract features of needle-shaped elongated weeds.
(4) The detection models established through different improvement methods underwent multiple experimental validations. Initially, various attention mechanisms and enhanced key modules were integrated into three different backbone networks. Subsequently, the performance of the improved models was tested on a weed dataset. Following this, ablation experiments were conducted to compare the effects of the three improvement methods on weed detection performance and select the optimal model suitable for the task. Finally, heatmap analysis was performed, visualizing feature maps using GradCam to analyze feature extraction.

2. Materials and Methods

2.1. Datasets

The paper selected three common types of weeds found in cornfields, with some of the images sourced from publicly available datasets [23]. Due to partial occlusion in some weed images within the original dataset and the presence of multiple images of the same weed, the training process may inadvertently learn repetitive features, leading to model overfitting. To address these issues, 200 high-quality images were selected for each weed type from the original dataset, with selection criteria including non-occluded weeds, varying shooting angles, and images captured at different times. Considering the limited number of samples, the paper further collected an augmented dataset of three weed types—Setaria viridis, Chenopodium, and Sonchus oleraceus—in cornfields. The weed images in the dataset are illustrated in Figure 1.

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

Figure 1. Pictures of three types of weeds in the dataset. (**a**) *Setaria viridis*; (**b**) *Chenopodium*; (**c**) *Sonchus oleraceus*.

The acquisition equipment used in this study is Redmi K40, 4800 pixels. The pictures of the three types of weeds were collected under the condition of a main view, side view, and top view, and pictures were taken at long range and close range. A 0.5× focal length in the camera was selected to simulate the long-range picture, and a 2× focal length was selected to simulate the close-range picture. In order to further simulate the influence of light conditions on the quality of picture shooting, a time-division shooting scheme was adopted. The time periods were selected from 8:00 to 10:00, 13:00 to 15:00, and 16:00 to 18:00, respectively. The three weeds were photographed, and the typical weed images collected are shown in Figure 2.

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

Figure 2. Collected pictures of three types of weeds. (**a**) *Setaria viridis*; (**b**) *Chenopodium*; (**c**) *Sonchus oleraceus*.

Since YOLOV8 requires supervised learning, the obtained dataset is labeled in Labelme [24] software, Version 5.1.1, which was developed by MIT's Computer Science and Artificial Intelligence Laboratory, and the results of the labeled JSON file are converted into a TXT file in YOLOv8 format to obtain the positional information of the weeds in the image. Figure 3 shows the distribution of position information of weed targets in the image in the dataset. From the figure, it can be seen that the weeds are widely distributed, and most of them are located in the center of the image. The horizontal and vertical coordinates in Figure 3 are the ratio of the height and width of the weed-labeling box to the height and width of the whole image, respectively, and the larger the value is, the larger the proportion of weeds in the image is. The image contains weed data of various sizes.

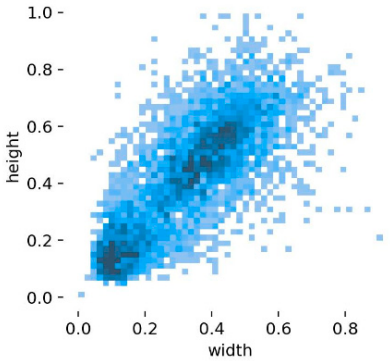

Figure 3. Distribution of weeds in the dataset.

2.2. YOLOv8 Object Detection Algorithm

There are five different versions of YOLOv8, distinguished by the number of modules and network layers, but their basic structure remains similar. The overall structure of YOLOv8 can be divided into the backbone network, neck, and head. The backbone is responsible for extracting features from the input image, which are then transmitted through the backbone network. The neck is for multi-scale feature fusion, and the head is for object detection, which includes predicting object location and category. The CBS module performs convolution, BN, and SiLU operations on the input image. The C2f structure draws inspiration from CSP and ELAN design principles, ensuring that the network is lightweight while obtaining rich gradient flow information. The SPPF module is responsible for converting feature maps of arbitrary sizes into feature maps of fixed sizes.

The model network architecture of the YOLOv8 target detection algorithm is depicted in Figure 4.

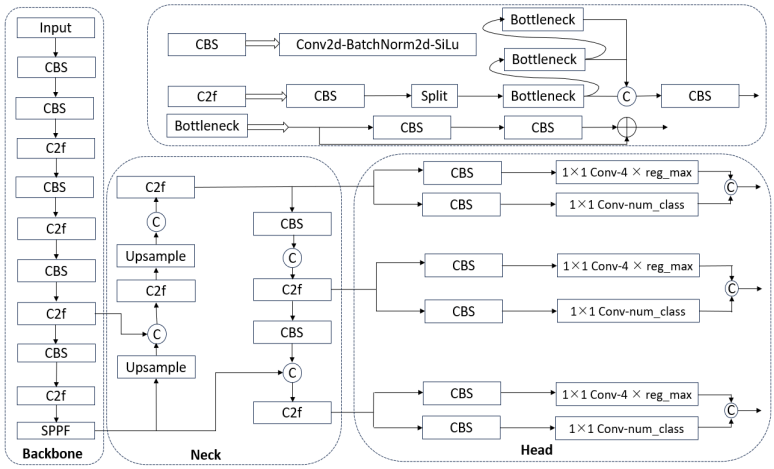

Figure 4. Overall structure of YOLOv8.

2.3. Backbone Network Improvements

To make the model more lightweight and improve its performance in dense weed detection, the original backbone network in YOLOv8 was replaced with RepViT and EfficientViT, respectively. The modified model backbone network architecture is depicted in Figure 5, with each red color representing a different backbone.

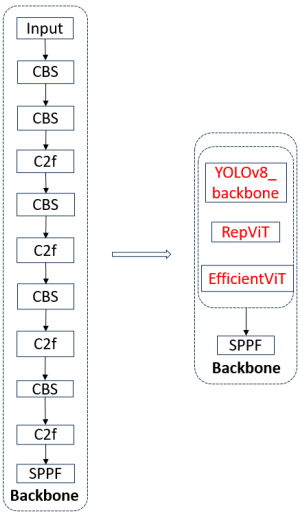

Figure 5. The overall structure of the YOLOv8 backbone network after improvement.

2.3.1. EfficientViT Dense Prediction Network

The Vision Transformer (ViT) [25] has found widespread application in computer vision tasks such as semantic segmentation, image classification, and object detection, owing to its multi-head self-attention mechanism, which facilitates the learning of global features. In order to further enhance the phenotypic performance of the model, the number of parameters in ViT has been progressively increased. However, this increase in parameters has led to a corresponding increase in model latency, rendering deployment on edge devices more challenging and unsuitable for real-time applications.

EfficientViT aims to improve the computational efficiency of ViT from three aspects: memory access, computation redundancy, and parameter efficiency. This is achieved by reducing the number of inefficient layers to enhance memory efficiency, slicing only half of the head input to reduce computational redundancy, and enhancing parameter efficiency through channel pruning to eliminate unimportant channels. To realize these objectives, EfficientViT is designed with cascaded group attention modules, memory-efficient sandwich layouts, and parameter redistribution strategies to enhance efficiency in terms of computation, memory, and parameters. The network structure of EfficientViT is illustrated in Figure 6, consisting of sandwich layouts and cascaded group attention modules. The EfficientViT series comprises six models with varying depths and width ratios, with a certain number of heads allocated for each stage. For further improvement in detection efficiency, EfficientViT-M0 is selected as the backbone network for weed detection tasks.

In weed detection tasks, individual weed growth is uncommon, whereas clustered or widespread growth is more typical. Mixed growth of different weed species in a given space, coupled with dense weed populations, poses a challenge to correctly identifying various weed types. To achieve accurate recognition and real-time detection of densely growing weeds while enhancing detection efficiency, the backbone network of YOLOv8 is improved to EfficientViT. Different attention mechanisms are then incorporated under the new backbone network to select the optimal improvement solution.

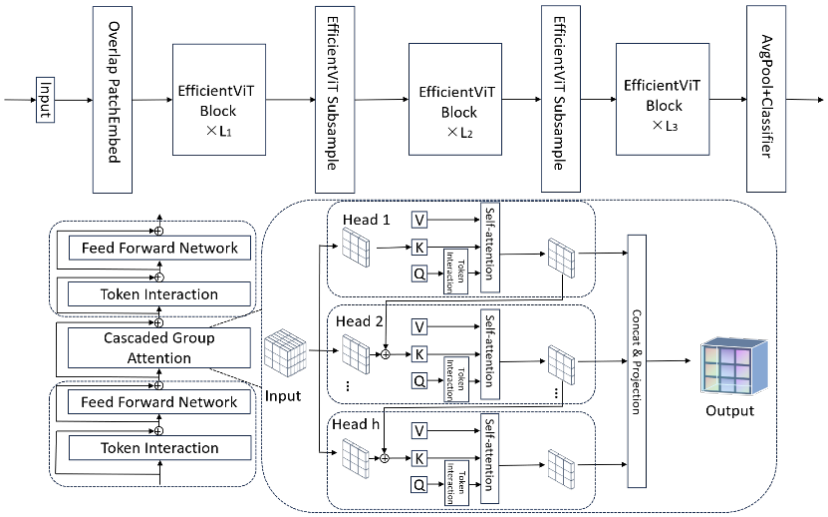

Figure 6. EfficientViT network structure.

2.3.2. RepViT Lightweight Network

Various efficient convolutional neural network attention mechanisms and different methods, such as designing convolutional kernels of different sizes, short-circuiting between modules, and stacking convolutions, are applied in networks to enhance performance in visual tasks. With the increase in network depth and stacking of various modules, the network's capability to represent features significantly improves. However, this improvement comes with negative impacts, such as a dramatic increase in computational complexity, leading to longer training times and higher inference latency, making it challenging to meet the real-time requirements of certain applications.

To reduce computational complexity, input features are often divided into multiple groups based on different computing devices, such as the Spatial Group-wise Enhance (SGE) [26] attention mechanism, which focuses on the feature factors of each subgroup. This method highlights the importance of each sub-feature, thereby enhancing the network's learning capability of different features. In convolutional neural networks, during feature extraction, emphasis is typically placed on fusing local features across all channels, neglecting potential interactions between features in different channels. The role of the channel mixer is to fuse features across different channels to make the extracted features richer and more diverse.

RepViT places depth-wise separable convolution modules at the beginning, followed by SE modules to separate the channel mixer and token mixer. During the training phase, an additional multi-branch depth-wise separable convolution structure is added to further enhance performance. In the inference phase, this branch is removed to reduce model latency. Connecting the aforementioned structure with the FFN network yields the RepViT block.

RepViT consists of three stages. The backbone network is composed of a stack of 3×3 convolutions with a stride of 2. Compared to the MobileNetV3 [27], although the number of convolution kernels has increased, the latency has decreased. The downsampling layer increases the network's depth and alleviates the loss of image information caused by the decrease in resolution during feature extraction. In RepViT, the deep downsampling layer comprises RepViT blocks, depth-wise separable convolutions, and FFNs. RepViT has five variants, namely, RepViT-M0.9/M1.0/M1.1/M1.5/M2.3, with different latencies determined by the number of channels used in each stage of the backbone network and the number of blocks in each block. In this improvement, RepViT-M0.9 is adopted.

In summary, RepViT is a novel lightweight network structure suitable for edge devices, achieving high accuracy while ensuring that the network remains lightweight. To further reduce the parameter count of YOLOv8, the original backbone network is replaced with RepViT, and the improved network structure is illustrated in Figure 7.

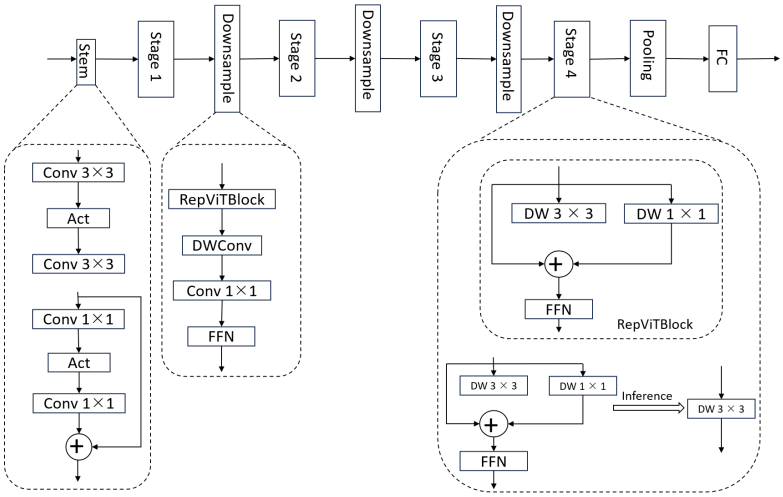

Figure 7. RepViT network architecture.

2.4. Key Module Improvements

The C2f module is one of the crucial components of the YOLOv8 network. To enhance the detection capability, particularly for flexible objects like needle-shaped weed leaves, improvements were made to both the C2f and Bottleneck modules in YOLOv8. Specifically, the original convolution computations in the C2f module were replaced with dynamic snake convolution.

Dynamic Snake Convolution

The shape and size of weed leaves vary, including scale-shaped, linear, thorn-shaped, and needle-shaped. When conducting specific weed detection, especially for features like needle-shaped leaves, feature maps are prone to cracks or discontinuities. This issue leads to lower accuracy during the inference phase. The original YOLOv8 model achieved only a 91.1% accuracy in detecting such weeds, with high rates of missed and false detections. To improve the accuracy of detecting these types of weeds, dynamic snake convolution was introduced on top of the original network.

The purpose of dynamic snake convolution is to enhance the recognition of curved and slender tubular structures. These structures, particularly the terminal branches of vessels, have numerous branches, thin vessel walls, and a complex, curved shape. They occupy fewer pixels in images, making it challenging for traditional convolution operations to extract their features completely. If the target for detection is within a complex background environment, with long structures and multiple branches, the model may overfit during training and cause false positives during inference, leading to decreased generalization of the model. Needle-shaped weed structures are similar to slender tubular structures, making it difficult to extract their features. Dynamic snake convolution adjusts the convolutional movement direction based on the previous convolution position, allowing it to capture richer feature information compared to traditional convolutions.

Though deformable convolutions [28] can learn offsets and adaptively adjust the kernel size to accommodate changes in object shape, it is challenging to focus on curved and slender structures. Dynamic snake convolution imposes purposive constraints on the process of changing the kernel size based on the shape of the object to be detected within deformable convolutions. It designs multiple convolutional kernel structures to supplement critical features from multiple perspectives, achieving efficient multi-perspective feature fusion to summarize typical feature learning.

Dynamic snake convolution combines traditional convolution with deformable convolution. For 2D convolution, with a given 3×3 convolutional kernel, the center coordinates can be represented as (x_i, y_i), and the standard convolutional kernel can be represented as

$$K = \{(x-1, y-1), (x-1, y), \ldots, (x+1, y+1)\} \tag{1}$$

Introducing an offset from the standard convolution kernel Δ, in order to prevent the receptive field from drifting off the target, constraints are imposed on the standard convolution kernel in both the x-axis direction and the y-axis direction:

$$K_{i\pm c} = \begin{cases} (x_{i+c}, y_{i+c}) = \left(x_i + c, y_i + \sum_i^{i+c} \Delta y\right), \\ (x_{i-c}, y_{i-c}) = \left(x_i - c, y_i + \sum_{i-c}^{i} \Delta y\right), \end{cases} \tag{2}$$

$$K_{j\pm c} = \begin{cases} (x_{j+c}, y_{j+c}) = \left(x_j + \sum_j^{j+c} \Delta x, y_j + c\right), \\ (x_{i-c}, y_{i-c}) = \left(x_j + \sum_{j-c}^{j} \Delta x, y_j - c\right), \end{cases} \tag{3}$$

Since the offsets are fractional, their integer coordinates are computed by bilinear interpolation, the $K = \sum K' B(K', K)$, which is a bilinear interpolating kernel function, and K' denotes the spatial position coordinate value.

The C2f structure is a crucial component of YOLOv8, playing a pivotal role in the network. Moreover, the design of the C2f module draws inspiration from the principles of the Bottleneck module. Therefore, improving the convolution computations in both the C2f and Bottleneck modules to dynamic snake convolution is significant. The modified model neck network architecture is depicted in Figure 8, with the red color representing the improvement of key modules.

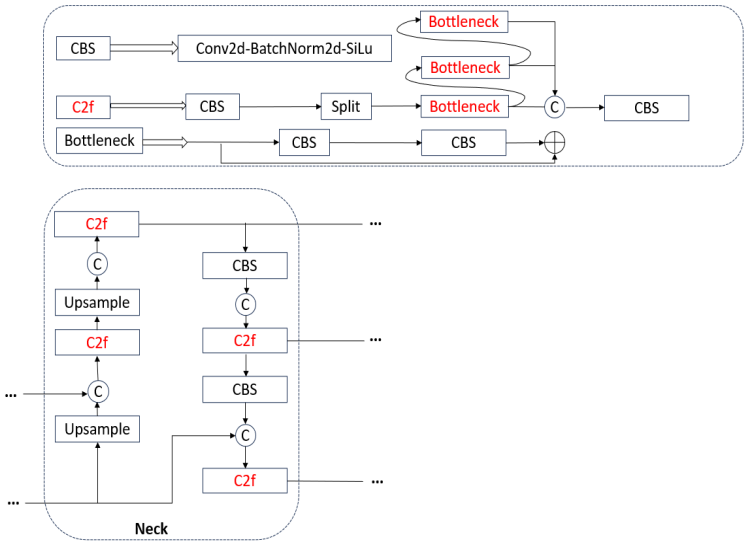

Figure 8. Diagram of the neck structure after improving the key modules.

2.5. Attention Mechanisms

In order to achieve the goal of efficient feature fusion for fast weed detection, various attention mechanisms were added to three different backbone networks to study their impact on recognition accuracy. The improved model with these added attention mechanisms is shown in Figure 9, with the red color representing the various attention mechanisms.

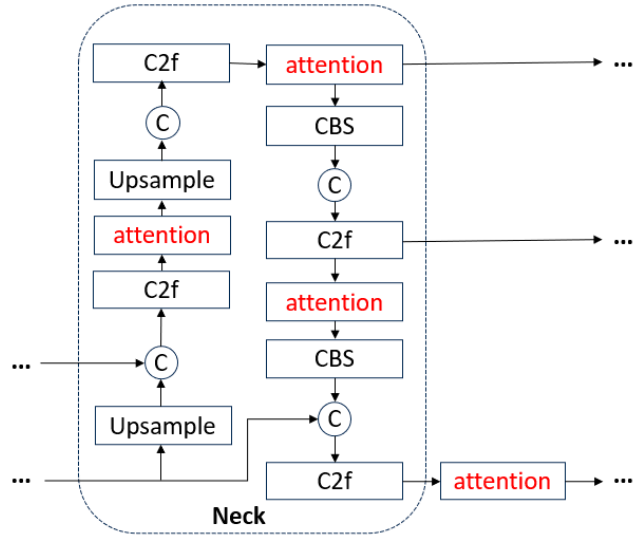

Figure 9. YOLOv8 neck structure after improving C2f module.

2.5.1. SimAM Attention Mechanism

For improving the performance ability of the convolutional neural network, on the one hand, various different modules can be designed from the network structure, such as Residual Unit [5], Inception [4], Dense [29] block, and so on. Humans can extract the important information contained in a picture at a glance when observing the picture, which is due to the fact that humans can focus their attention on the important positions in the picture. Different attention mechanisms have been designed based on the attention mechanism in human visual information processing. Methods are based on the attention mechanism, due to its flexibility and plug-and-play features that can be used in different positions in a convolutional neural network. Channel attention and spatial attention focus on different things: the former focuses more on the extraction of features in small regions of the image and assigns a weight coefficient to that feature for each channel during computation, whereas the latter will assign a larger weight coefficient to the focal region. To summarize, channel attention determines the specific location of the target object in the image, and spatial attention achieves the purpose of determining the type of the object. Both of these attentional mechanisms are similar to the feature-based and spatial-based attentional mechanisms in the human brain, but the two attentional mechanisms in the human brain have a certain interaction, and when observing an object, human beings can independently select the features related to the observed object and ignore the irrelevant information. Under the joint action of the two, humans can quickly extract the information in the object that is useful to them. Current attention mechanisms do not take into account the effect of the joint action of the two on the features, such as CBAM [30], which is mixed after calculating one- and two-dimensional weights, respectively. In order to combine the weights calculated by channel attention and spatial attention, SimAM starts from human neurology theory [31], calculates the weights occupied by different neurons, i.e., calculates an energy function based on the importance of neurons, and calculates 3D weights based on the closed-form solution of the energy function.

$$e_t(w_t, b_t, y, x_i) = (y_t - \hat{t})^2 + \frac{1}{M-1}\sum_{i=1}^{M-1}(y_0 - \hat{x}_i)^2 \tag{4}$$

$$e_t(w_t, b_t, y, x_i) = \frac{1}{M-1}\sum_{i=1}^{M-1}(-1 - (w_t x_i + b_t))^2 + (1 - (w_t t + b_t))^2 + \lambda {w_t}^2 \tag{5}$$

where w_t and b_t are the weights and biases, t denotes the target neuron, and x_i denotes the other neurons.

There should be M energy functions on each channel, and the analytical solution of the above equation is

$$w_t = -\frac{2(t-\mu_t)}{(t-\mu_t)^2 + 2\sigma_t^2 + 2\lambda} \tag{6}$$

$$b_t = -\frac{1}{2}(t+\mu_t)w_t \tag{7}$$

where $\mu_t = \frac{1}{M-1}\sum_{i=1}^{M-1} x_i$, ${\sigma_t}^2 = \frac{1}{M-1}\sum_{i=1}^{M-1}(x_i - \hat{\mu}_t)^2$

For this mean and variance on the channel, it can be seen from Equations (6) and (7) that the analytic solution is obtained on a single channel, and therefore, it is inferred that other neurons on the same channel also satisfy the same distribution. Therefore, all neurons on the same channel can reuse this mean and variance.

Minimum energy is obtained in the end:

$${e_t}^* = \frac{4\left(\hat{\sigma}^2 + \lambda\right)}{(t-\hat{\mu})^2 + 2\hat{\sigma}^2 + 2\lambda} \tag{8}$$

where $\hat{\mu} = \sum_{i=1}^{M} x_i$, $\hat{\sigma}^2 = \sum_{i=1}^{M}(x_i - \hat{\mu})^2$

In the energy function (1), the linear separability between the target neuron and the surrounding neurons is first calculated, and an active neuron will exhibit inhibitory properties to other surrounding neurons, i.e., the neuron should be assigned a higher weight value, where Equations $\hat{t} = w_t t + b_t$ and, $x_i = w_t x_i + b_t$, $M = H \times W$ denote the number of neurons on a given channel.

The smaller the energy ${e_t}^*$, the more inhibition it exhibits to peripheral neurons, i.e., the higher the importance of the neuron in the visual processing task—the degree of importance can be expressed as $1/{e_t}^*$, so with a larger $1/{e_t}^*$—the higher the importance, and when the opposite occurs, the lower the importance. Attentional modulation in the mammalian brain manifests itself as a gain (i.e., scaling) effect on neuronal responses, using scaling operators for feature refinement:

$$\widetilde{X} = sigmoid\left(\frac{1}{E}\right) \odot X \tag{9}$$

where E is the energy function of each neuron, sigmoid restricts E to larger values, and X is the input feature.

2.5.2. EMA Attention Mechanism

In the CA (Coordinate Attention) [32] mechanism, position information is embedded to establish a link between the channel and the space, and the horizontal and vertical position information is aggregated by pooling the input features in parallel global averaging, although the different directional position information retained by this method can capture spatially remote interactions, as well as the whole spatial positional interactions of the features. The main improvements of the EMA attention module are as follows:

(1) Feature grouping: The input feature map is divided into G sub-features in order to learn different semantics;
(2) Parallel sub-network: The parallel sub-network is divided into three branches, namely two 1×1 branches and one 3×3 branch. The former aggregates horizontal and vertical position coding through global average pooling, and the latter captures multi-scale features;
(3) Cross-space learning: The outputs of 1×1 and 3×3 in the parallel sub-network are aggregated across spatial information in different spatial dimensional direc-

tions to realize different scales of feature fusion and capture more comprehensive contextual information.

$$Z_c^W(W) = \frac{1}{H} \sum_{0 \le j \le H} x_c(j, W) \tag{10}$$

where $Z_c^W(H)$ and $Z_c^W(W)$ is the set of position information along the vertical and horizontal directions, respectively, C is the number of input channels, and H and W are the height and width of the input feature maps, respectively. The structure of the EMA module is shown in Figure 10.

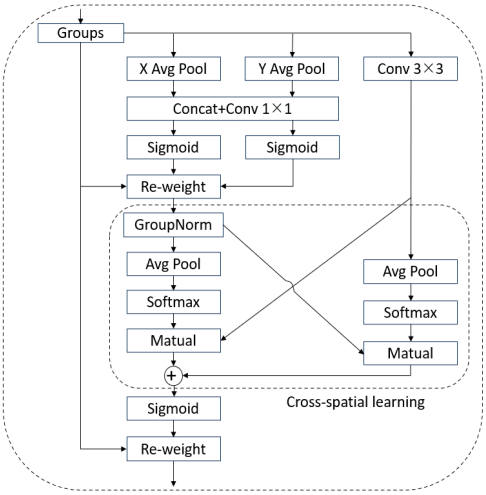

Figure 10. EMA module structure.

2.5.3. BiFormer

The self-attention mechanism in Transformer can capture long-range dependency, but this approach leads to increased computation and high memory usage; BiFormer implements dynamic sparse attention to remove irrelevant information through Bi-Layer Routing Attention (BRA) for dynamic computational allocation and content awareness.

BiFormer's features are summarized below.

(1) Divide the input feature map into $S \times S$ lattices, and then obtain Q, K, and V by linear mapping.

$$Q = X^r W^q, K = X^r W^k, V = X^r W^v \tag{11}$$

(2) Average the Q and K of each region to obtain the regional average values of Q^r and K^r; construct a directed graph to obtain the region in which each key–value pair in each region should be involved by means of the adjacency matrix, where the adjacency matrix can be expressed as below.

$$A^r = Q^r (K^r)^T \tag{12}$$

(3) Calculate the routing index matrix I^r by the adjacency matrix to obtain the most relevant top k regions for each region to participate in the fine-grained operation as relevant regions.

$$I^r = topkIndex(A^r) \tag{13}$$

(4) Since the routing regions are scattered throughout the feature map and are difficult to compute, they are first aggregated.

$$K^g = gather(K, I^r) \tag{14}$$

$$V^g = gather(V, I^r) \tag{15}$$

(5) The aggregated K^g, V^g pairs use the attention operation to obtain the final attention.

$$O = Attention(Q, K^g, V^g) + LCE(V) \tag{16}$$

where $LCE(V)$ parameterizes the depth convolution.

The BiFormer block is constructed by BRA, and the final BiFormer is obtained by stacking this module. The specific structure of the BiFormer block and BiFormer is shown in Figure 11.

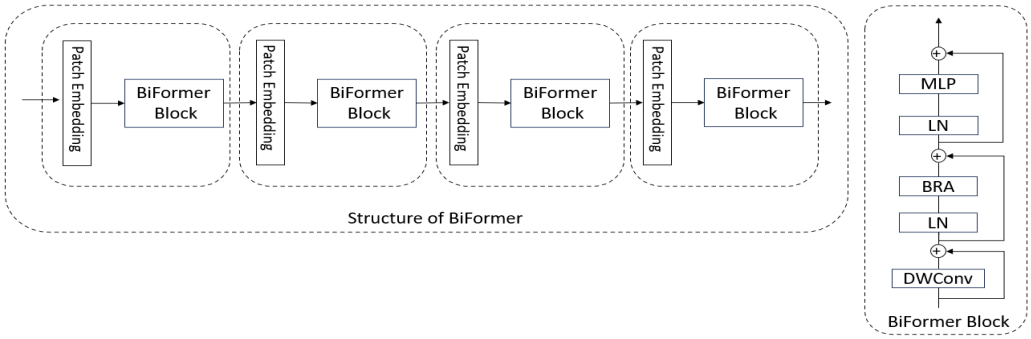

Figure 11. BiFormer module structure.

2.6. Experiments

2.6.1. Experiment Protocol

(1) To further validate the improved network's ability to enhance performance in weed detection tasks, the improved network is first subjected to comparative experiments with the original network.

(2) Under the condition of equal training parameters, the improved backbone network is combined with various attention mechanisms and enhancements to important modules. The network structure of the improved model is illustrated in Figure 12, with the red color representing the three methods of improvement.

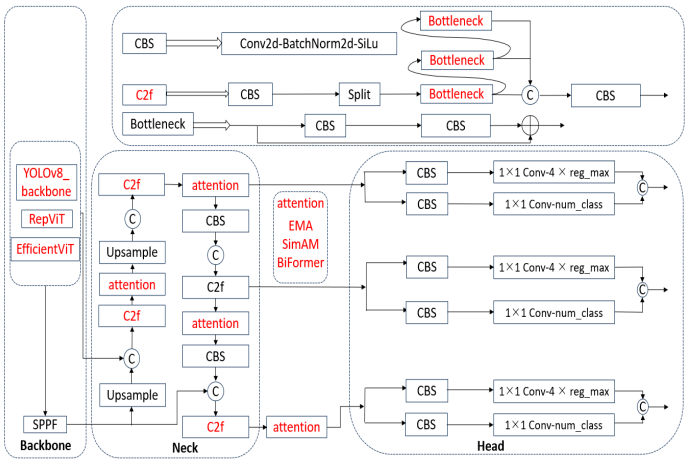

Figure 12. YOLOv8 weed detection model structure with different improvement methods.

(3) Following the comparison of individually enhanced models, Section 3 of this chapter will conduct ablation experiments to assess the extent to which different combinations of improvement methods affect weed detection performance. The optimal improvement method will be selected for application in weed detection tasks. This

experiment is conducted on a Windows 10 Professional system, with PyTorch version 2.1.0 and CUDA version 12.1.0. Specific configurations are detailed in Table 1, and the hyperparameters for training the improved model are outlined in Table 2.

Table 1. System configuration.

Configure	Version/Model
CPU	Intel Xeon W-2223
GPUs	GTX3090
CUDA	12.1
Pytorch	2.1.0

Table 2. Experimental parameter configuration.

Epoch	300
Batch	16
Learning Rate	0.01
IOU	0.7
Classes	3

2.6.2. Evaluation Metrics

In order to accurately evaluate the performance of the model before and after the improvement, *Precision*, *Recall*, *Average Precision* (AP) and *mean Average Precision* (mAP) are used as evaluation metrics.

$$Precision = \frac{TP}{TP + FP} \times 100\% \tag{17}$$

$$Recall = \frac{TP}{TP + FN} \times 100\% \tag{18}$$

$$AP = \int_0^1 P(R)dR \tag{19}$$

$$mAP = \frac{\sum AP}{K_{Class}} \tag{20}$$

where *TP* represents the number of positive samples predicted as positive, *FP* represents the number of negative samples predicted as positive, *FN* represents the number of positive samples predicted as negative, and *TN* represents the number of negative samples predicted as negative. *Precision* represents the proportion of samples predicted as positive and correctly predicted among all samples predicted as positive. *Recall* represents the proportion of samples predicted as positive among all positive samples in the dataset. The *mAP* is the mean of the *Average Precision (AP)* and the *Average Precision (AP)* is the area of the P-R curve.

2.6.3. Results

To validate the performance of the models with the addition of the SimAM and EMA attention mechanisms, dynamic snake convolution, as well as the improvement of converting the backbone network into EfficientViT and RepViT, experiments were conducted on the weed dataset. The results are presented in Tables 3–5. It can be observed that under the original YOLOv8 backbone network, dynamic snake convolution exhibited the most significant improvement in model performance. Precision, recall, mAP (0.5), and mAP (0.5:0.95) increased by 5.6%, 5.8%, 6.4%, and 1%, respectively. By adding dynamic snake convolution, the next convolution position could freely choose the direction based on the previous convolution kernel. This allows for the extraction of more feature information, particularly for flexible and slender weeds, thereby reducing the false-negative rate.

Table 3. Improvement and experimental data with the original backbone network.

Improved Methodology	Accuracy/%	Recall Rate/%	mAP(0.5)%	mAP(0.50:95)/%
YOLOv8	91.1	88.4	90.5	83.3
DCNv3	91.9	89.3	96.2	83.7
DySkD	96.7	94.2	96.9	84.3
SimAM	95.6	90.7	96.4	82.5
EMA	96.2	91.1	96.2	82.3
BiFormer	95.0	93.9	95.7	83.1
YOLOv8_DSEB	95.9	94.3	96.5	83.8

Table 4. Improvements and experimental data with EfficientViT backbone network.

Improved Methodology	Accuracy/%	Recall Rate/%	mAP(0.5)/%	mAP(0.5:0.95)/%
EfficientViT	96.0	93.7	96.5	84.0
DCNv3	96.5	91.8	95.6	83.4
DySkD	96.3	92.5	96.5	83.4
SimAM	97.0	92.6	96.4	83.3
EMA	96.7	92.1	96.6	83.8
BiFormer	97.4	93.1	96.5	84.1
YOLOv8_EDSEB	96.7	93.1	96.4	83.3

Table 5. Improvement and experimental data with RepViT backbone network.

Improved Methodology	Accuracy/%	Recall Rate/%	mAP(0.5)/%	mAP(0.5:0.95)/%
RepViT	96.1	92.3	96.0	83.1
DCNv3	96.5	91.8	95.6	83.4
DySkD	96.9	92.5	96.6	83.4
SimAM	96.0	93.9	96.4	83.3
EMA	96.3	92.2	96.5	83.3
BiFormer	96.5	93.7	96.3	83.2
YOLOv8_RDSEB	95.9	93.3	96.5	83.4

EfficientViT integrates multi-scale attention, allowing for a global receptive field and enabling the prediction of dense weeds. As shown in the Table 4, when the backbone network is upgraded to EfficientViT, the missed detection rate of dense weeds can be effectively reduced. Particularly, when BiFormer is added to the EfficientViT backbone network, the model's performance improves significantly, with precision, recall, mAP (0.5), and mAP (0.5:0.95) increasing by 6.3%, 4.7%, 6%, and 0.8% respectively. Adding BiFormer not only enhances the detection capability of dense weeds but also improves their localization accuracy.

Under the RepViT backbone network, the impact of dynamic snake convolution and deformable convolution on model performance improvement is comparable. Though the overall performance of the model is not as strong as the other two enhancement methods, there is still an improvement compared to before the enhancements. Particularly, with the addition of the SimAM attention mechanism, the Recall rate increases significantly. The SimAM attention mechanism, similar to human observation of objects, facilitates interaction between channel attention and spatial attention, thus enhancing localization and classification accuracy. Consequently, it can reduce the missed detection rate, with precision, recall, and mAP (0.5) improving by 5.9%, 5.5%, and 5.9% respectively.

3. Ablation Experiments

3.1. Analysis of Ablation Experiments Results

To compare the effects of improving the backbone network, enhancing key modules, and adding different attention mechanisms on weed detection performance, and to select the optimal improvement method suitable for weed detection tasks, a total of fifteen experiments were designed to assess the impact of different improvement methods on weed detection performance. These experiments were conducted on the same weed dataset, and the results of the ablation experiments are presented in Tables 6–8.

Table 6. Ablation experiments under the original backbone network.

Improvement Methods	Precision/%	Recall/%	mAP(0.5)/%	mAP(0.5:0.95)%
YOLOv8	91.1	88.4	90.5	83.3
YOLOv8 + Dynamic snake	96.7	94.2	96.9	84.3
YOLOv8 + Dynamic snake + SimAM	97.0	93.3	96.8	84.2
YOLOv8 + Dynamic snake + EMA	95.6	94.2	96.7	83.9
YOLOv8 + Dynamic snake +BiFormer	95.1	93.3	96.2	83.6

Table 7. Ablation experiments under the EfficientViT backbone network.

Improvement Methods	Precision/%	Recall/%	mAP(0.5)/%	mAP(0.5:0.95)%
EfficientViT	96.0	93.7	96.5	84.0
EfficientViT + Dynamic snake	96.3	92.5	96.5	83.4
EfficientViT + Dynamic snake + SimAM	97.1	94.3	96.9	84.1
EfficientViT + Dynamic snake + EMA	97.3	92.7	96.6	83.6
EfficientViT + Dynamic snake + BiFormer	95.6	92.8	96.6	83.6

Table 8. Ablation experiments under the RepViT backbone network.

Improvement Methods	Precision/%	Recall/%	mAP(0.5)/%	mAP(0.5:0.95)%
RepViT	96.1	92.3	96.0	83.1
RepViT + Dynamic snake	96.9	92.5	96.6	83.4
RepViT + Dynamic snake + SimAM	97.0	92.4	96.3	83.4
RepViT + Dynamic snake + EMA	95.4	93.5	96.2	83.2
RepViT + Dynamic snake + BiFormer	95.1	93.3	96.2	83.6

From the results of the ablation experiments, it can be observed that all improvement methods led to performance enhancements compared to the original network. Under the original backbone network and the improved EfficientViT and RepViT backbone networks, adding dynamic snake convolution and the SimAM attention mechanism effectively enhanced the model's detection capability. Compared to individually adding dynamic snake convolution and the SimAM attention mechanism under the three backbone networks, the combination of the two further improved performance. Dynamic snake convolution autonomously selects the direction of convolution, and the SimAM attention mechanism integrates channel and spatial attention, further strengthening the detection capability of small targets, thus reducing both missed detections and false positives. Therefore, the combination of dynamic snake convolution and SimAM is considered the optimal improvement method.

Under the original backbone network, adding dynamic snake convolution and the SimAM attention mechanism resulted in improvements of 5.9% in precision, 4.9% in recall, 6.3% in mAP (0.5), and 0.9% in mAP (0.5:0.95). Under the EfficientViT backbone network, adding dynamic snake convolution and the SimAM attention mechanism led to improvements of 6% in precision, 5.9% in recall, 6.4% in mAP (0.5), and 0.7% in mAP (0.5:0.95). Under the RepViT backbone network, adding dynamic snake convolution and

the SimAM attention mechanism resulted in improvements of 5.9% in precision, 4% in recall, 5.8% in mAP (0.5), and 1% in mAP (0.5:0.95).

Based on the comprehensive analysis, selecting EfficientViT as the backbone network, enhancing the key module with dynamic snake convolution, and adding the SimAM attention mechanism is deemed the optimal model for weed detection tasks. This model is named EDS-YOLOv8, denoting the optimal improvement model.

3.2. Heatmap Analysis

To provide a more intuitive visualization of the improved YOLOv8's detection performance, GradCam [33] heatmaps were utilized to visualize the detection results, as shown in Figure 13. The original YOLOv8 object detection model and the enhanced EDS-YOLOv8 weed detection model were selected for visualization using GradCam. From Figure 13, it can be observed that though the YOLOv8 backbone network primarily focuses attention on the objects to be detected, it pays less attention to the leaves. After improving the backbone network to EfficientViT, the model can detect small targets and dense weeds as well. With the addition of dynamic snake convolution, the model can capture the fine structures of weed leaves and extract their features, effectively reducing missed detections of small targets. Furthermore, the incorporation of the SimAM attention mechanism strengthens the fusion between channel and spatial attention features, enabling precise localization and classification.

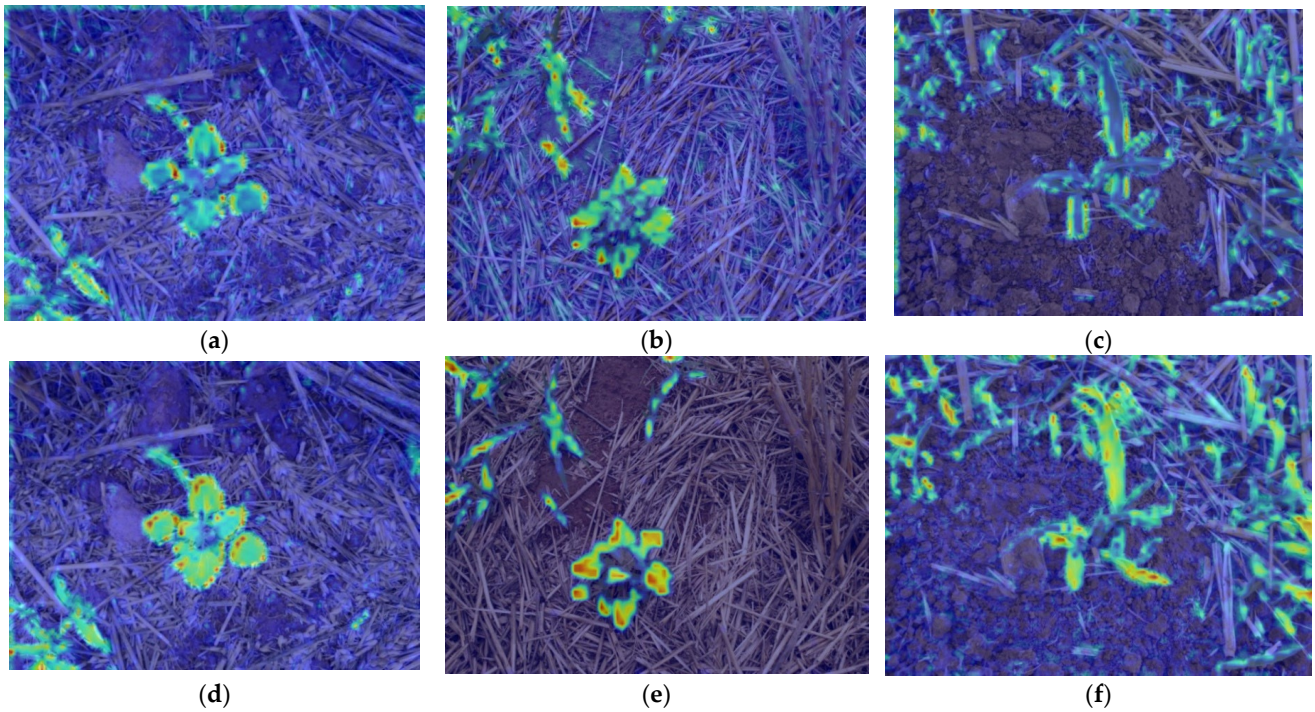

Figure 13. YOLOv8 heat map and EDS-YOLO heat map comparison. (**a**–**c**) YOLOv8 heat map; (**d**–**f**) EDS-YOLO thermogram.

4. Conclusions

By introducing dynamic snake convolution, the model can adaptively learn the size of convolutional kernels, thus enhancing the detection capability of flexible objects. Additionally, the incorporation of the SimAM, EMA, and BiFormer attention mechanisms in the backbone network further strengthens the feature fusion capability. To further ensure that the model is lightweight for deployment on edge devices, this paper introduces the RepViT lightweight backbone model, onto which various modules are added. In order to achieve dense weed detection, the EfficientViT dense prediction network is introduced, along with different attention mechanisms. Notably, adding dynamic snake convolution to the

original backbone network shows the most significant improvement in model performance, with precision, recall, mAP (0.5), and mAP (0.5:0.95) increasing by 5.6%, 5.8%, 6.4%, and 1%, respectively.

From the results of the ablation experiments, it is evident that stacking modules does not significantly improve model performance. It is crucial to choose attention mechanisms tailored to specific tasks. Given the unique shape of weed leaves, characterized as flexible objects, dynamic convolution autonomously learns the size of convolutional kernels. It supplements important features from multiple perspectives, facilitating comprehensive feature summarization learning. This significantly enhances model performance, making dynamic snake convolution particularly suitable for weed detection tasks. Through heatmap analysis, the improved network directs more attention to the objects to be detected, thereby increasing the accuracy of weed detection, especially for needle-shaped weeds. Compared to the original algorithm, the improved YOLOv8 algorithm achieves a 6% increase in precision, a 5.9% increase in recall, and improvements of 6.4% and 0.7% in mAP (0.5) and mAP (0.5:0.95), respectively. This enhancement enhances YOLOv8's ability to detect flexible objects, meeting the requirements of weed detection tasks and making the model well suited for field detection tasks.

Through ablation experiments and heatmap analysis, the effectiveness of each improvement in enhancing model performance was verified. Additionally, the optimal model suitable for weed detection tasks was selected, which involves improving the backbone network to EfficientViT and adding dynamic snake convolution and the SimAM attention mechanism.

Author Contributions: C.H.: Conceptualization, Methodology, Validation, Program modification, Formal analysis, Data curation, Writing—original draft preparation, Writing—review and editing, Visualization. F.W.: Methodology, Investigation, Writing—review and editing. G.M.: Methodology, Investigation, Writing—review and editing. X.M.: Validation, Resources, Project administration, Funding acquisition. K.Z.: Methodology, Investigation, Writing—review and editing. X.W.: Investigation, Project administration. X.H.: Validation, Resources, Supervision, Project administration, Formal analysis, Funding acquisition. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the National Natural Science Foundation of China (No. 32160426) and Key Research and Development Projects in Gansu Province (No. 23YFNA0014).

Institutional Review Board Statement: Not applicable.

Data Availability Statement: The datasets generated for this study are available upon request from the corresponding author.

Acknowledgments: We thank the scientific research team at Gansu Agricultural University for agricultural mechanization and automation and for help and encouragement.

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

References

1. Girshick, R. Fast R-CNN. In Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV), Santiago, Chile, 7–13 December 2015; pp. 1440–1448.
2. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. ImageNet Classification with Deep Convolutional Neural Networks. *Commun. ACM* **2017**, *60*, 84–90. [CrossRef]
3. Simonyan, K.; Zisserman, A. Very Deep Convolutional Networks for Large-Scale Image Recognition. *arXiv* **2014**, arXiv:1409.1556. [CrossRef]
4. Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P.; Reed, S.; Anguelov, D.; Erhan, D.; Vanhoucke, V.; Rabinovich, A. Going Deeper with Convolutions. *arXiv* **2014**, arXiv:1409.4842. [CrossRef]
5. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. *arXiv* **2015**, arXiv:1512.03385. [CrossRef]
6. 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]
7. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You Only Look Once: Unified, Real-Time Object Detection. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 779–788.

8. Redmon, J.; Farhadi, A. YOLO9000: Better, Faster, Stronger. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; IEEE Computer Society: Los Alamitos, CA, USA, 2017; pp. 6517–6525.
9. Redmon, J.; Farhadi, A. YOLOv3: An Incremental Improvement. *arXiv* **2018**, arXiv:1804.02767.
10. Bochkovskiy, A.; Wang, C.-Y.; Liao, H.-Y.M. YOLOv4: Optimal Speed and Accuracy of Object Detection. *arXiv* **2020**, arXiv:2004.10934.
11. Jocher, G.; Chaurasia, A.; Stoken, A.; Borovec, J.; Kwon, Y.; Michael, K.; Fang, J.; Yifu, Z.; Wong, C.; Montes, D.J.Z.; et al. Ultralytics/Yolov5: V7.0—YOLOv5 SOTA Realtime Instance Segmentation 2022. Available online: https://ui.adsabs.harvard.edu/abs/2022zndo...3908559J/abstract (accessed on 19 April 2024).
12. Li, C.; Li, L.; Jiang, H.; Weng, K.; Geng, Y.; Li, L.; Ke, Z.; Li, Q.; Cheng, M.; Nie, W.; et al. YOLOv6: A Single-Stage Object Detection Framework for Industrial Applications. *arXiv* **2022**, arXiv:2209.02976.
13. 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. In Proceedings of the 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Vancouver, BC, Canada, 17–24 June 2023; pp. 7464–7475.
14. Jocher, G.; Chaurasia, A.; Qiu, J. Ultralytics YOLO 2023. Available online: https://github.com/ultralytics/ultralytics (accessed on 19 April 2024).
15. Wang, C.-Y.; Yeh, I.-H.; Liao, H.-Y.M. YOLOv9: Learning What You Want to Learn Using Programmable Gradient Information. *arXiv* **2024**, arXiv:2402.13616.
16. Xu, K.; Li, H.; Cao, W.; Zhu, Y.; Chen, R.; Ni, J. Recognition of Weeds in Wheat Fields Based on the Fusion of RGB Images and Depth Images. *IEEE Access* **2020**, *8*, 110362–110370. [CrossRef]
17. Xu, X.H.; Quan, H.R.; He, K.Y.; Wang, L.; Wang, X.Q.; Wang, Q. Proportional Fluorescence Sensing Analysis of Pesticide Residues in Agricultural Environment. *Trans. Chin. Soc. Agric. Mach.* **2020**, *51*, 229–234.
18. Wang, B.J.; Lan, Y.B.; Chen, M.M.; Liu, B.H.; Wang, G.B.; Liu, H.T. Application Status and Prospect of Machine Learning in Unmanned Farm. *J. Chin. Agric. Mech.* **2021**, *42*, 186–192+217. [CrossRef]
19. Yuan, T.; Hu, T.; Ma, C.; Li, L.Y.; Zheng, X.G.; Qian, D.L. Weed Target Detection Algorithm in Paddy Field Based on YOLOv4. *Acta Agric. Shanghai* **2023**, *39*, 109–117. [CrossRef]
20. Coates, A.; Ng, A.Y. Learning Feature Representations with K-Means. In *Neural Networks: Tricks of the Trade*; Lecture Notes in Computer Science; Springer: Berlin/Heidelberg, Germany, 2012; Volume 7700, pp. 561–580.
21. Cao, Y.L.; Zhao, Y.W.; Yang, L.L.; Li, J.; Qin, L.L. Weed Ldentification Method in Rice Field Based on Improved DeepLabv3+. *Trans. Chin. Soc. Agric. Mach.* **2023**, *54*, 242–252.
22. Shang, W.Q.; Qi, H.B. Identification Algorithm of Field Weeds Based on Improved Faster R-CNN and Transfer Learning. *J. Chin. Agric. Mech.* **2022**, *43*, 176–182. [CrossRef]
23. Jiang, H.; Zhang, C.; Qiao, Y.; Zhang, Z.; Zhang, W.; Song, C. CNN Feature Based Graph Convolutional Network for Weed and Crop Recognition in Smart Farming. *Comput. Electron. Agric.* **2020**, *174*, 105450. [CrossRef]
24. Wada, K. Labelme: Image Polygonal Annotation with Python. Available online: https://github.com/wkentaro/labelme (accessed on 19 April 2024).
25. Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; et al. An Image Is Worth 16x16 Words: Transformers for Image Recognition at Scale. *arXiv* **2020**, arXiv:2010.11929. [CrossRef]
26. Li, X.; Hu, X.; Yang, J. Spatial Group-Wise Enhance: Improving Semantic Feature Learning in Convolutional Networks. *arXiv* **2019**, arXiv:1905.09646. [CrossRef]
27. Howard, A.; Sandler, M.; Chu, G.; Chen, L.-C.; Chen, B.; Tan, M.; Wang, W.; Zhu, Y.; Pang, R.; Vasudevan, V.; et al. Searching for MobileNetV3. *arXiv* **2019**, arXiv:1905.02244. [CrossRef]
28. Dai, J.; Qi, H.; Xiong, Y.; Li, Y.; Zhang, G.; Hu, H.; Wei, Y. Deformable Convolutional Networks. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 764–773.
29. Huang, G.; Liu, Z.; van der Maaten, L.; Weinberger, K.Q. Densely Connected Convolutional Networks. *arXiv* **2016**, arXiv:1608.06993. [CrossRef]
30. Woo, S.; Park, J.; Lee, J.-Y.; Kweon, I.S. CBAM: Convolutional Block Attention Module. *arXiv* **2018**, arXiv:1807.06521. [CrossRef]
31. Webb, B.S.; Dhruv, N.T.; Solomon, S.G.; Tailby, C.; Lennie, P. Early and Late Mechanisms of Surround Suppression in Striate Cortex of Macaque. *J. Neurosci. Off. J. Soc. Neurosci.* **2005**, *25*, 11666–11675. [CrossRef] [PubMed]
32. Qiao, S.; Shen, W.; Zhang, Z.; Wang, B.; Yuille, A. Deep Co-Training for Semi-Supervised Image Recognition. *arXiv* **2018**, arXiv:1803.05984. [CrossRef]
33. Selvaraju, R.R.; Cogswell, M.; Das, A.; Vedantam, R.; Parikh, D.; Batra, D. Grad-Cam: Visual Explanations from Deep Networks via Gradient-Based Localization. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 618–626.