

Article

# Real-Time Detection of *Eichhornia crassipes* Based on Efficient YOLOV5

Yukun Qian<sup>1</sup>, Yalun Miao<sup>1</sup>, Shuqin Huang<sup>1</sup>, Xi Qiao<sup>2</sup> , Minghui Wang<sup>1</sup>, Yanzhou Li<sup>1,\*</sup>, Liuming Luo<sup>1</sup>, Xiyong Zhao<sup>1,2</sup> and Long Cao<sup>1</sup><sup>1</sup> School of Mechanical Engineering, Guangxi University, Nanning 530004, China<sup>2</sup> Shenzhen Branch, Guangdong Laboratory of Lingnan Modern Agriculture, Genome Analysis Laboratory of the Ministry of Agriculture and Rural Affairs, Agricultural Genomics Institute at Shenzhen, Chinese Academy of Agricultural Sciences, Shenzhen 518120, China

\* Correspondence: lyz197916@126.com

**Abstract:** The rapid propagation of *Eichhornia crassipes* has a threatening impact on the aquatic environment. For most small water areas with good ecology, daily manual monitoring and salvage require considerable financial and material resources. Unmanned boats have important practical significance for the automatic monitoring and cleaning *Eichhornia crassipes*. To ensure that the target can be accurately detected, we solve the problems that exist in the lightweight model algorithm, such as low accuracy and poor detection effect on targets with small or unclear characteristics. Taking YOLOV5m 6.0 version as the baseline model, given the computational limit of real-time detection, this paper proposes to use EfficientNet-Lite0 as the backbone, use the ELU function as the activation function, modify the pooling mode in SPPF, embed the SA attention mechanism, and add the RFB module in the feature fusion network to improve the feature extraction ability of the whole model. The dataset collected water hyacinth images from ponds and lakes in Guangxi, Yunnan, and the China Plant Image Library. The test results show that efficient YOLOV5 reached 87.6% mAP, which was 7.1% higher than that of YOLOV5s, and the average detection time was 62 FPS. The ablation experiment verifies the effectiveness of each module of efficient YOLOV5, and its detection accuracy and model parameters meet the real-time detection requirements of the *Eichhornia crassipes* unmanned cleaning boat.

**Keywords:** *Eichhornia crassipes* detection; YOLOV5; deep learning; EfficientNet

**Citation:** Qian, Y.; Miao, Y.; Huang, S.; Qiao, X.; Wang, M.; Li, Y.; Luo, L.; Zhao, X.; Cao, L. Real-Time Detection of *Eichhornia crassipes* Based on Efficient YOLOV5. *Machines* **2022**, *10*, 754. <https://doi.org/10.3390/machines10090754>

Academic Editor: Xiangjun Zou

Received: 6 July 2022

Accepted: 8 August 2022

Published: 1 September 2022

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

*Eichhornia crassipes* originated in northeastern Brazil in South America is now widely distributed in North America, Asia, Oceania, Africa, and many other countries. It is one of the most harmful aquatic floating plants on earth and can grow indefinitely under suitable conditions, with a rapid propagation speed [1,2]. *Eichhornia crassipes* population outbreaks can reduce water quality, affect biodiversity, and destroy ecosystem functions [3–5]. In Dianchi Lake, Yunnan Province, 60% of the local species were extinct, due to the invasion of water hyacinth [6]. To prevent the irreparable loss of the ecological system that is caused by the mass reproduction of *Eichhornia crassipes*, frequent manual cleaning is needed. According to the obtained data, 10 and 5 million yuan are spent annually in Wenzhou City, Zhejiang Province, and Putian City, Fujian Province to salvage *Eichhornia crassipes*. In Lavaping Lake in central Java, Indonesia, to control the growth of water hyacinth, excavators, multipurpose dredgers, amphibious weed harvesters, and other machines are used for daily cleaning, and the annual consumption is as high as 500 million Indonesian shields [7]. Traditional water hyacinth treatments include physical, chemical, and biological methods. The physical method uses an artificial machine to clean or set fences to prevent the drift of *Eichhornia crassipes*. Zhang Li et al. [8] and Zhang Zhiyong et al. [9] controlled the reproduction of *Eichhornia crassipes* by designing anti-wind

fences. The scheme is effective but produces a considerable economic burden. Zheng Bin et al. [10] used chemical methods to prevent and control the reproduction of *Eichhornia crassipes* and proposed that the use of Clerodendrum trichotomum extract can make water hyacinth leaves dry, rot, and die. However, chemical treatment methods result in pollution and damage to the water ecological environment. Biological methods use the mutual restriction of organisms to inhibit the extensive propagation of *Eichhornia crassipes*. There is a hidden danger of introducing other exotic species, and the effect is slow. It is difficult to reduce the damage caused by the invasion of *Eichhornia crassipes* in a short time [11].

With the increasingly rigorous ecological environment, it is urgent to protect the safety of the water resource ecological system. The cleaning of *Eichhornia crassipes* has attracted increasing attention from scholars. Most scholars have used remote sensing technology to assist in formulating cleaning schemes. Ling Sun et al. [12] and Ling Zhou et al. [13] proposed constructing a logistic model to simulate the diffusion process of water hyacinths. Furthermore, they obtained the spatiotemporal diffusion model of *Eichhornia crassipes* using differential equations. J.A. Mukarugwiro et al. [14] and Abeyou W. Worqlul et al. [15] collected multispectral image data over the years by satellite to analyse the range and distribution of invasive alien weeds over time. Ling Sun et al. [16] and Timothy Dube et al. [17] used multiple sensors to collect information, and discriminant analysis (DA) and partial least squares discriminant analysis (PLS-DA) were used to obtain the key information needed for the water hyacinth eradication scheme. Kgabo H. Thamaga et al. [18] realized the identification, detection, and plotting of the spatial distribution configuration of invasive *Eichhornia crassipes* in small reservoirs using a combination of the Landsat 8 land imager (OLI) and Sentinel-2 multispectral instrument (MSI). Previous scholars mostly analysed the situation of large water areas but did not fully understand the distribution of *Eichhornia crassipes* in small freshwater systems, such as lakes, streams, and rivers. At the same time, they lacked suitable satellite platforms and multispectral equipment that could distinguish the water hyacinth from another plant species [19]. In most small freshwater systems with better ecological environments, daily manual monitoring and fishing are only performed to prevent the sudden emergence and rapid reproduction of *Eichhornia crassipes*, which results in environmental pollution and consumes considerable human and financial resources. In 2020, Zhongao Feng et al. [20] proposed using machine vision and image recognition technology to design an intelligent monitoring device for the growth of water hyacinths and identify their growth state and scope, as well as to assist in the prevention and control of agriculture. However, the scheme designed by the scholar can only realize real-time monitoring in a small water space, which is limited. In recent years, with the rapid development of machine vision, deep learning, and image processing technology, the lightweight target detection algorithm has been realized. The monitoring and fishing tasks of *Eichhornia crassipes* in small water systems will tend to be automated, and the research and development of *Eichhornia crassipes* unmanned cleaning boats are necessary.

Currently, the target detection algorithm based on deep learning has developed more maturely. Due to the different detection frameworks, target detection technology is divided into one and two stages. The two-stage network obtains the approximate location, size, and other information of the target through the network regression of the first stage. In the second stage, the size, location, and category information of the target box are obtained through the detection head, such as Fater-RCNN [21] networks and other improved versions. The core idea of the one-stage detection algorithm is that the input image is directly regressed through the network to obtain the size, location, and category information of the target box. Although the detection accuracy of the one-stage algorithm is generally worse than that of the two-stage algorithm, the detection speed and training time of the one-stage algorithm is shorter than those of the two-stage algorithm. For example, the YOLO [22] algorithm has an excellent effect on real-time detection and recognition accuracy. Its improved versions YOLOV3 [23], YOLOV4 [24], and YOLOV5 combine various means and skills to further promote the detection performance of the algorithm.

The one-stage algorithm has been applied to target detection tasks in various areas of life. For real-time detection problems, JongBae Kim [25] and Luyao Du et al. [26] used image brightness equalization, noise removal, and other image dataset preprocessing methods to improve the accuracy of real-time detection of vehicles and lights. Xuelong Hu et al. [27] proposed improving the connection mode of PAN in the aggregation network in the feature fusion network based on the YOLOV4 algorithm. DenseNet was used to modify the residual connection mode in the CSP (cross-stage partial networks) structure, so that the average accuracy of real-time detection of underwater feed was improved from 65.4% to 92.61%. To achieve the real-time detection of wild animals on the CPU platform, Tingting Zhao et al. [28] used MobileNet to replace the backbone feature extraction network in YOLO. After the experiments, the average accuracy was increased by 7.7%, and the FPS (frames per second) were increased by 3 percent. Yuhao Ge et al. [29] used Shufflenetv2 and CBAM modules on the backbone of YOLOV5s model and BiFPN in the neck network. Under the condition of 10.5 M parameters, the detection accuracy of flowers, green tomatoes, and red tomatoes were 93.1%, 96.4%, and 97.7%, respectively. Wang Xuewen et al. [30] proposed a lightweight, small object, real-time detection architecture LDS-YOLO and designed a new feature extraction module. This module reuses features from front layers to achieve dense connections, thus reducing the dependence on datasets. Finally, the AP is 89.11%, and the number of parameters is only 7.6 M.

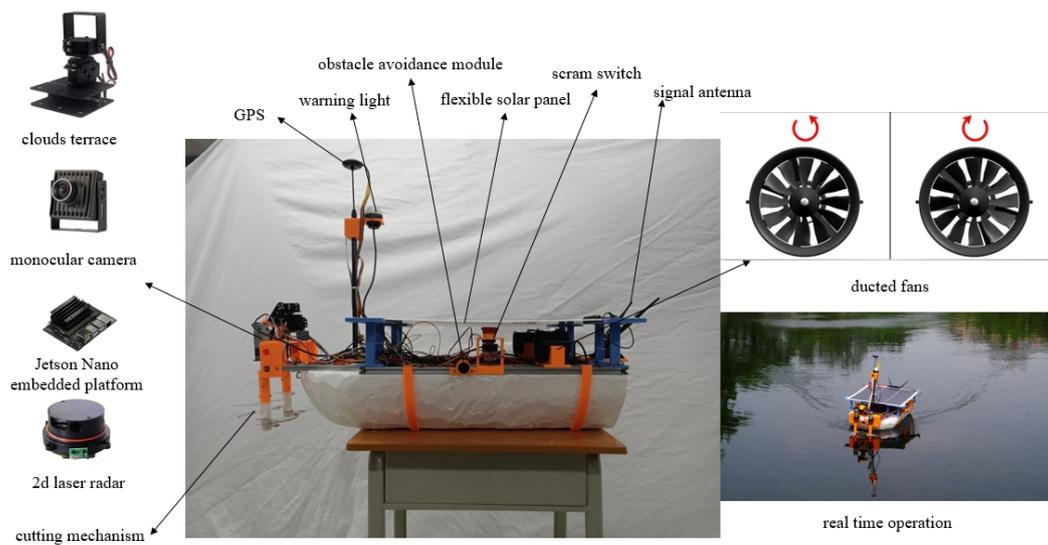
With the continuous development of deep learning networks and demand for lightweight models in the use of scenes, the abovementioned methods cannot meet the requirements of high precision and FPS real-time detection of *Eichhornia crassipes* targets with variable shapes in feature extraction and feature fusion. In this study, the YOLOV5 6.0 model was modified to design a network model that can be used on an embedded platform and meet the real-time and accurate detection of *Eichhornia crassipes* in different, complex, small water area environments by unmanned *Eichhornia crassipes* cleaning boats. The contributions of this work are as follows.

- (1) In the feature extraction network, with EfficientNet-Lite0 as the backbone, the activation function was modified from ReLU6 (rectified linear unit 6) to ELU (exponential linear units), and MaxPool in SPPF (spatial pyramid pooling fast) was modified to SoftPool and embedded into the SA (shuffle attention) attention mechanism module to enhance the local information extraction ability of the algorithm.
- (2) In the feature fusion network, the original FPN (feature pyramid networks) and PAN (path aggregation network) were used as the baseline models, and the RFB (receptive field block net) module was added to strengthen the extraction ability of network features.
- (3) Efficient YOLOV5 algorithm was proposed for real-time detection. The detection effect of different aggregation forms of *Eichhornia crassipes* in different environments was better, and the detection accuracy reached 87.6%. Experiments show that efficient YOLOV5 has the best comprehensive performance in terms of accuracy, detection speed, and generalization ability.

## 2. Methods

### 2.1. Introduction of Unmanned Boats

The unmanned cleaning boat of *Eichhornia crassipes* mainly consists of five parts: the energy, target detection, automatic control, navigation, and cutting parts. As shown in Figure 1, the target detection part of the unmanned boat is composed of a Jetson-nano embedded platform, monocular camera, and stable gimbal. The monocular camera adopted an RGB three-color camera with a USB interface protocol, and its resolution was  $640 \times 480$ . The camera was placed on the front stable gimbal, and its vision can be left or right  $\pm 30^\circ$ . The automatic control part adopted the STM32F407 control platform. When working, the control platform received the target direction returned by the embedded platform and coordinate data returned by the laser radar, adjusted the hull position by the PID (process identification) control method, and navigated to the target for cutting. To ensure the real-time accuracy of *Eichhornia crassipes* detection, this study evaluated the visual algorithm.



**Figure 1.** Unmanned *Eichhornia crassipe* cleaning boat.

## 2.2. Overview of the Network Model

Efficient YOLOV5 uses the latest YOLOV5 6.0 as the baseline model. Considering that the model applies to the embedded platform with limited computation, it needs to achieve higher accuracy with less computation. The YOLOV5 algorithm was divided into four versions (i.e., s, m, l, and x) according to the width and depth of the network model. Comparing the training of YOLOV5s with YOLOV5m, from the data in Table 1 was found that the model parameters and floating-point calculations trained by YOLOV5s were small, which can meet the demand of real-time detection, though the accuracy was slightly lower. The model trained by YOLOV5m had good accuracy, but it was difficult to achieve the real-time detection ability in an embedded platform because of the large amount of required calculations.

**Table 1.** Comparison of the training results of YOLOV5s and YOLOV5m.

Model	Depth Multiple	Width Multiple	Params	GFLOPs	mAP@50	mAP@.5:.95	FPS
YOLOV5s	0.33	0.50	6.69 M	15.8	0.805	0.412	92
YOLOV5m	0.67	0.75	19.89 M	48	0.836	0.505	48

Based on the abovementioned comparison, it was determined that the depth and width of the model have a great influence on the detection accuracy in the real-time detection of *Eichhornia crassipes*. To ensure higher accuracy under the premise of real-time detection, the improvement idea of efficient YOLOV5 is to take YOLOV5m as the baseline model, replace the feature extraction network of the original model with the lightweight network EfficientNet-Lite0, increase the attention mechanism, and improve the feature fusion network based on FPN and PAN. In the YOLOV5-6.0 version, the author deleted the previously used focus module and changed the SPP (spatial pyramid pooling) module to the SPPF module.

As shown in Figure 2, the specific process of the efficient YOLOV5 algorithm is as follows: the input image enters the backbone network to produce three feature maps with different scales, EfficientNet-Lite0 is the backbone network, and the stem module defines its activation function as ELU. There are 16 MBConv convolution blocks in the backbone network, followed by the SPPF module. To reduce the loss of information, the pooling mode of SPPF was modified to SoftPool, and the ultralightweight attention mechanism SA-Net integrating airspace and channel attention was added at the end of the backbone network. In the feature fusion network, efficient YOLOV5 improved the shortcomings of the original FPN and PAN

by strengthening the information flow. The receptive field block (RFB) module was added to the original top-down and cotton-up branches to enhance the multiscale modelling ability of the feature fusion network. Finally, a lightweight model that can realize real-time detection and higher precision on an embedded platform was obtained.

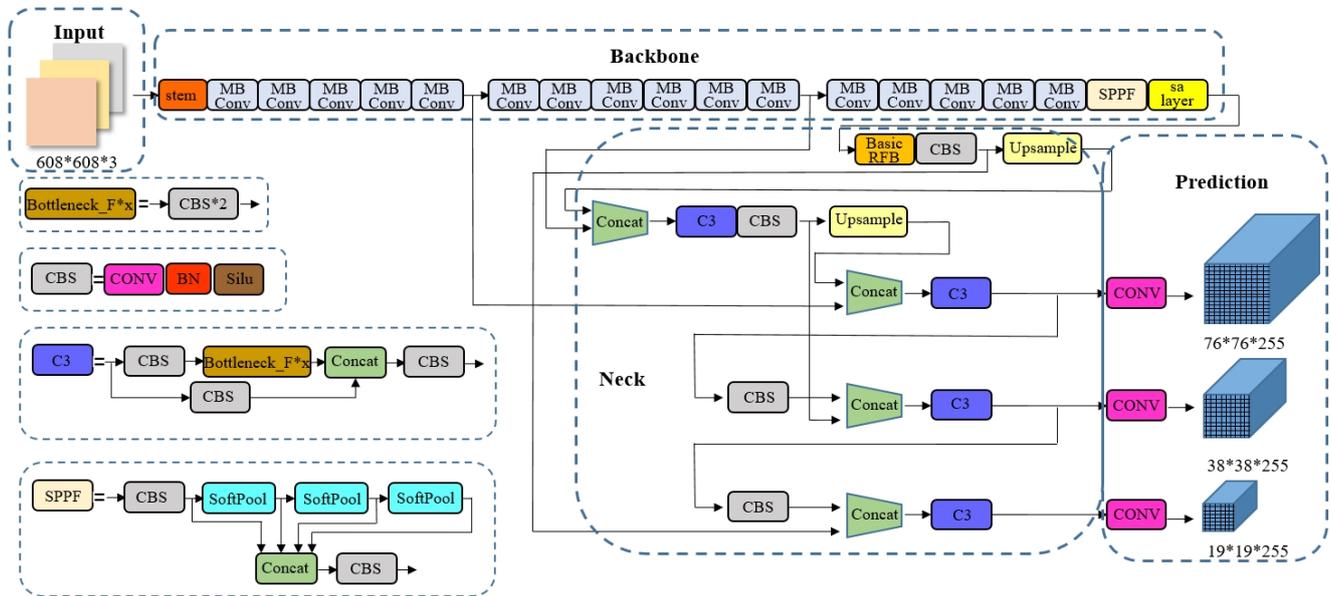


Figure 2. Efficient YOLOV5 model structure.

### 2.3. Feature Extraction Network

#### 2.3.1. EfficientNet-Lite0

The EfficientNet-Lite0, proposed by Google, was used as the baseline model for feature extraction, which was improved based on the EfficientNet classification and recognition network proposed by Tan et al. [31] in 2019. The basic idea of EfficientNet is to expand the three dimensions of network width, depth, and resolution of the input image, according to the proportion through the composite scaling method, in order to seek the optimal parameters needed to maximize the recognition accuracy, as shown in Figure 3e. The optimal combination was found without changing the number of model parameters. The author proposed a compound scaling method, which uses the mixed factor  $\phi$  to unify the scaling factors. The specific formulas are as follows:

$$\begin{cases} \text{depth} : d = \alpha^\phi \\ \text{width} : w = \beta^\phi \\ \text{resolution} : r = \gamma^\phi \end{cases} \quad (1)$$

$$\text{s.t.} \begin{cases} \alpha \cdot \beta^2 \cdot \gamma^2 \approx 2 \\ \alpha \geq 1, \beta \geq 1, \gamma \geq 1 \end{cases} \quad (2)$$

$\alpha$ ,  $\beta$ , and  $\gamma$  represent the distribution coefficients of network depth, width, and resolution, respectively, which can be obtained and fixed by standard network search.  $\phi$  is the composite coefficient, whose value is adjusted according to available resources. EfficientNet-B0 consists of 16 MBConv (mobile inverted bottleneck convolution), including deep separable convolution, Swish activation function dropout layer, and the SE (squeeze-excitation) attention mechanism module. The EffientNet-Lite0 used in this paper was improved, based on EfficientNet-B0. Specifically, the SE module was removed, and ReLU6 was used to replace the Swish activation function to fix the stem and head modules at the same time, when the model size was scaled to reduce the calculation amount. The specific structure of MBConv is shown in Figure 4.

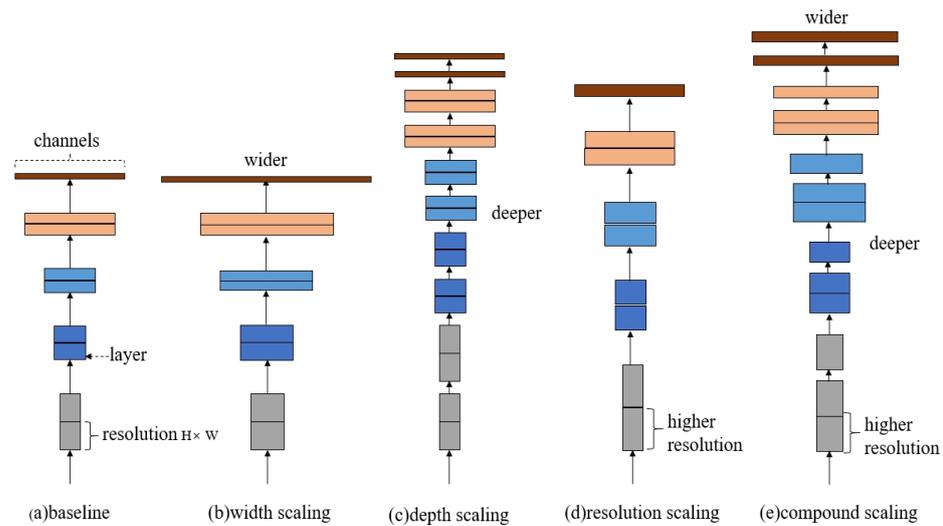


Figure 3. EfficientNet schematic.

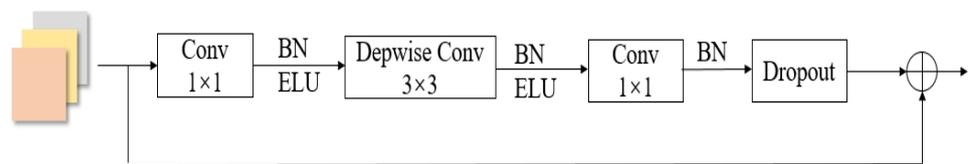


Figure 4. MBConv structure.

### 2.3.2. ELU

In efficient YOLOV5, the activation function of EfficientNet-Lite is modified from ReLU6 to ELU, its function image is shown in Figure 5. The purpose of the ReLU6 function is to prevent the loss of accuracy. It has a small amount of calculation and fast convergence speed. However, there may be problems, such as neuron death, and the gradient is always 0 during the training process. The formula is as follows (3). To solve this problem, we use the ELU activation function proposed by Djork-Arne Clevert et al. [32], which combines sigmoid and ReLU. The linear part on the right side of the ELU is the same as the ReLU activation function to alleviate the gradient disappearance. At the same time, the soft saturation on the left side can make the ELU more robust to input changes or noise. The main advantage is that there is no problem with neuronal death. When the input is small, it will be saturated to a negative value, thereby reducing the variation and information of forwarding propagation. The formula is as follows (4).

$$ReLU6(x) = \min(\max(0, x), 6) \tag{3}$$

$$ELU(x) = \begin{cases} x, & x > 0 \\ a(e^x - 1), & x \leq 0 \end{cases} \tag{4}$$

### 2.3.3. SoftPool

SPPF is a pooling method for multiscale feature fusion. The previous SPP module needs to specify the size of three convolution kernels. After three pooling and splicing, SPPF specifies a convolution kernel, and the output after each pooling will become the input of the next pooling, which can make the calculation speed faster.

The commonly used pooling methods include max and average pooling. The main role is to retain the main features and reduce the amount of calculation, but both methods have the problem of losing the details of the image when pooling. SoftPool [33] can minimize the loss of information in the process of pooling, while maintaining the function of the pooling

layer, with low computational and memory consumption, as well as a good detection effect for small targets. Therefore, SoftPool is selected here to replace MaxPool in SPPF.

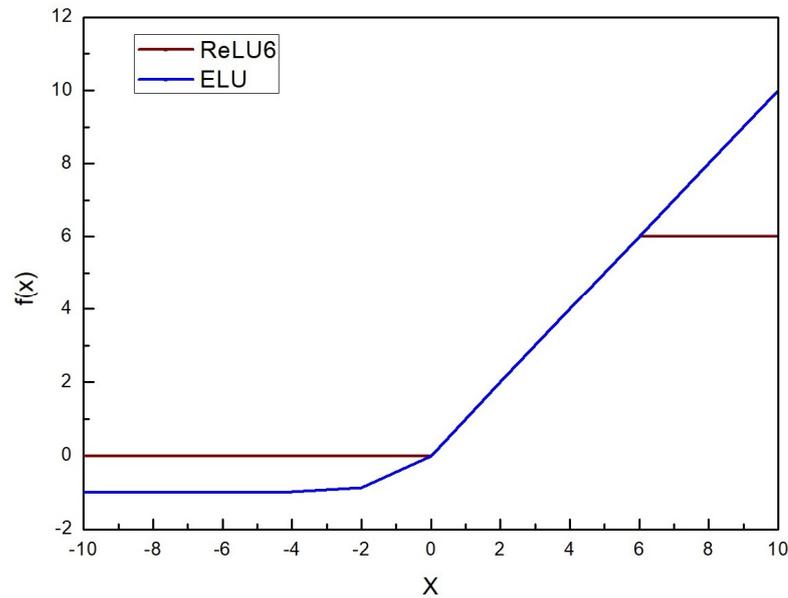


Figure 5. ELU and ReLU6 function graph.

SoftPool retains the input feature attribute by the Softmax weighting method. In the activation region, each activation factor obtains different weights, and the weighted summation of activation factors in the pooling kernel is realized by nonlinear transformation.  $a_i$  is the activator,  $W_i$  is the weight, which is calculated as the ratio of the natural index of the activator to the sum of the natural indices of all activators in the region, and  $\tilde{a}$  is the output value, which is obtained by weighted summation of all activation factors in the kernel domain.

$$W_i = \frac{e^{a_i}}{\sum_{i \in R} W_i \times a_i} \tag{5}$$

$$\tilde{a} = \sum_{i \in R} W_i \times a_i \tag{6}$$

Figure 6 is the bidirectional propagation process of SoftPool, and the region of  $6 \times 6$  size is represented as activation map (a), where the forward calculation includes calculating the weight  $W_i$  of the candidate  $3 \times 3$  region and multiplying the weight  $W_i$  and activation map (a) to obtain  $\tilde{a}$ . The inverse calculation includes calculating the gradient value  $\nabla \tilde{a}$  and multiplying it by the weight to obtain  $\nabla a_i$ .

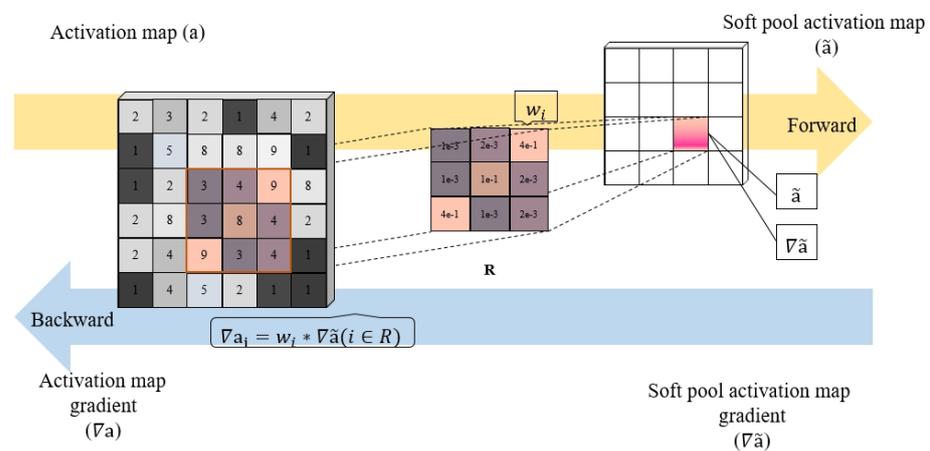


Figure 6. SoftPool schematic.

### 2.3.4. Shuffle Attention

The attention mechanism is a bionic visual mechanism that invests more attention resources in regions of interest to improve the efficiency and accuracy of visual information processing. The shuffle attention (SA) [34] mechanism, based on spatial and channel attention, introduces feature grouping and channel replacement to increase the sensitivity of the model to the characteristics of *Eichhornia crassipes*, as shown in Figure 7.

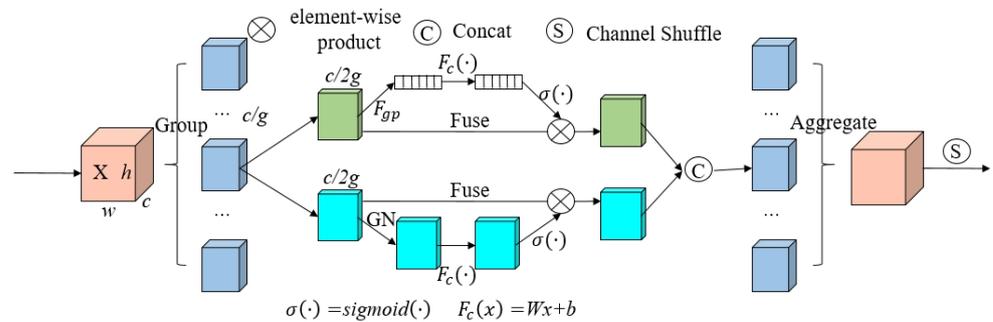


Figure 7. Shuffle attention schematic.

Set the input feature  $X \in R^{C \times H \times W}$  and split the input into  $g$  groups in the feature grouping step. Each group of features needs to generate different importance coefficients through spatial and channel attention modules. The upper half of the ink-green color is divided into channel attention. To minimize the weight, SA uses the simplest combination of GAP, scale, and Sigmoid. The lower part of light blue is the realization of spatial attention. The group norm (GN) is used to process  $X_{k2}$ , in order to obtain statistical information at the spatial level, and then  $F_C(\cdot)$  is used to enhance it. In the aggregate step, it is fused to obtain  $X'_k = [X'_{k1}, X'_{k2}] \in R^{C/2G \times H \times W}$ . Finally, the channel shuffle is used for intergroup communication, and the output is the same size as the input.

### 2.4. Feature Fusion Network

In the feature fusion network, YOLOV5 adopts the pattern of FPN and PAN, which enhances the local modelling ability of the feature hierarchy by establishing a bottom-up feature fusion network. However, the FPN and PAN models do not consider the semantic information difference between different layers of feature fusion, and there is a problem of multiscale information loss. To compensate for this defect, the RFB [35] module is added, and the residual branch is used to add different levels of spatial context information to the top-down branch, in order to reduce the loss of context information in the high-level feature map, as shown in Figure 8.

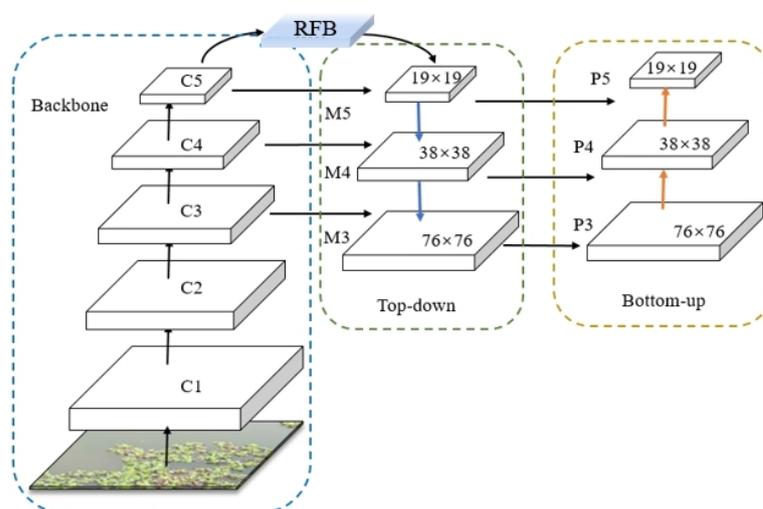
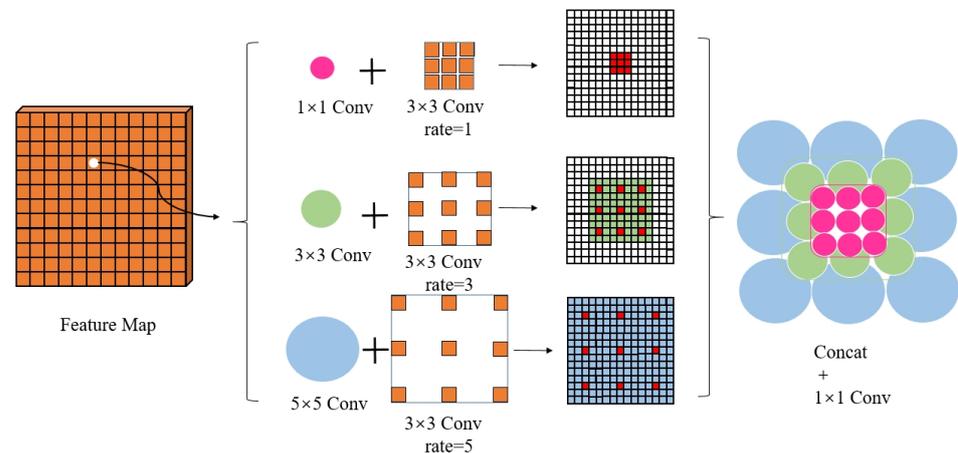


Figure 8. Feature fusion network structure.

The starting point of the RFB module simulates the receptive field of human vision to strengthen the feature extraction ability of the network. The specific method is to add cavity convolution to the inception. As shown in Figure 9, the convolution layers of convolution cores with different sizes constitute a multibranch structure, and the dilated convolution layer is introduced. The convolution layer outputs of different sizes and proportions are merged, and each branch covers different regions, which are fused to provide multiscale feature information.



**Figure 9.** Receptive field block schematic.

### 3. Experiments and Results

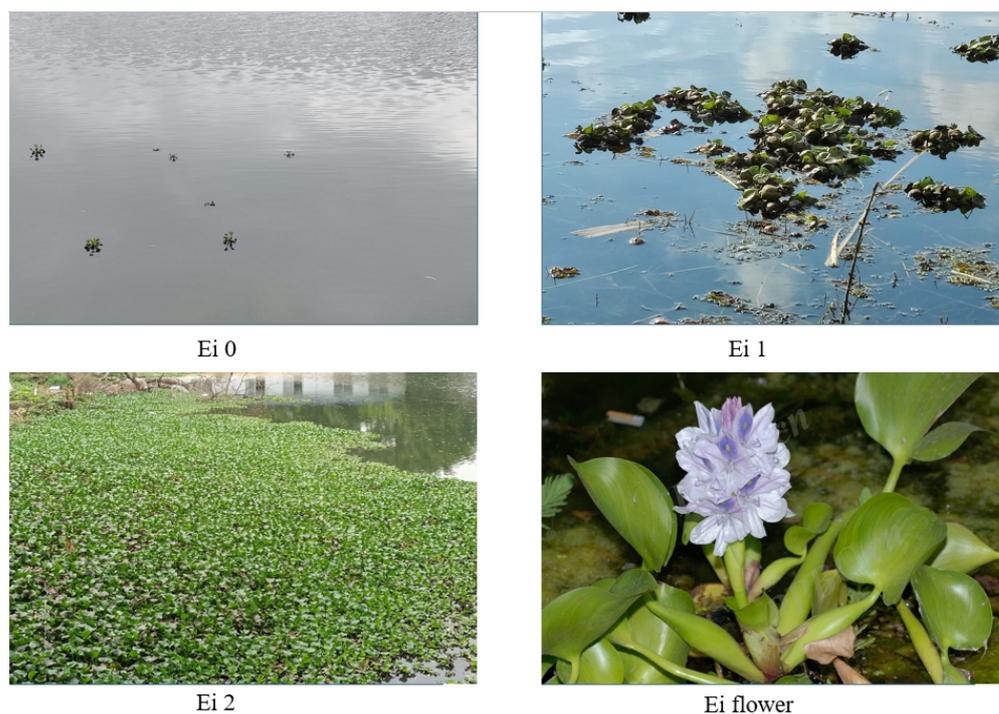
#### 3.1. Implementation Details

##### 3.1.1. Dataset Production

A total of 870 images of *Eichhornia crassipes* in different scenes are used as training datasets. The dataset includes pictures of *Eichhornia crassipes* flowers collected from the Chinese plant image database and photos of *Eichhornia crassipes* growth in different ponds and lakes. In this experiment, the image labelling software, i.e., makesense.ai, was used to label the image at the pixel level. The labelling data included the category and location information of the detected object, and 7571 labels were obtained. The picture collection and labels were made into COCO datasets. Due to the different aggregation forms of *Eichhornia crassipes* and large changes in characteristics, in order to ensure the accuracy of detection and facilitate the next cleaning, the collected images of *Eichhornia crassipes* were divided into four categories. The four categories are: Ei 0 *Eichhornia crassipes* individual form; Ei 1 *Eichhornia crassipes* aggregation form, which is also the most common form of *Eichhornia crassipes* growth; Ei 2 *Eichhornia crassipes* extremely dense form; and Ei flower *Eichhornia crassipes* flower morphology, as shown in Figure 10. The dataset was divided into training and testing sets, according to the proportions of 0.9 and 0.1.

##### 3.1.2. Training and Inference Details

Based on the PyTorch deep learning framework, in the comparison and ablation experiments, the baseline model was YOLOV5–6.0, backbone network was EfficientNet-Lite0, and neck network was FPN and PAN. The learning rate was set to 0.01, and the weight attenuation was 0.0005. By default, one GPU was used to train the model for 200 epochs. The experimental environments were Ubuntu18.04, PyTorch1.9, CUDA11.6, and cuDNN v8.2. The CPU version was Intel Xeon Sliver 4144, and the video card was P4000. The indicators in the COCO detection and evaluation standard were used for performance evaluation, including the average accuracy (mAP) of 0.5 for the IoU threshold, average accuracy (mAP @.5:.95) at different thresholds (from 0.5 to 0.95, step size 0.05), number of parameters (params), floating-point arithmetic (GFLOPs), and number of frames per second (FPS).



**Figure 10.** Different categories of *Eichhornia crassipes*.

### 3.2. Result Comparison

We compared efficient YOLOV5 with mainstream target detection algorithms, such as Faster RCNN and SSD. The experimental results are shown in the abovementioned Table 2. The average accuracy of the two-stage algorithm was higher in different thresholds, but the long training time and slow detection speed did not meet the requirements of real-time monitoring. The YOLOV4-tiny algorithm only used a feature pyramid in the feature fusion layer and abandoned the Mish activation function. The simple model and the minimum number of parameters make it the highest FPS value, in comparison to various algorithms, which meets the requirements of the embedded platform. However, the mAP values of the YOLOV4-tiny and SSD algorithms were too low to ensure the accuracy of detection in practical applications. As shown in Figure 11, the closer the algorithm was to the upper left corner, the better its performance. Efficient YOLOV5 appeared in the upper left corner, and its parameters were also acceptable in the range of the embedded platforms. The detection accuracies of various categories of *Eichhornia crassipes* were the highest. Overall, it can meet the design goal of real-time detection of *Eichhornia crassipes* by unmanned boats.

**Table 2.** Comparison of various detection algorithms.

Model	mAP@50	mAP@.5:.95	Params	GFLOPs	FPS	AP <sub>50</sub> of Classes			
						Ei 0	Ei 1	Ei 2	Ei Flower
Faster-RCNN	0.840	0.564	41.14 M	193.79	7	0.879	0.831	0.699	0.953
SSD	0.517	0.276	25.06 M	31	40	0.475	0.564	0.261	0.818
YOLOV4-tiny	0.505	0.254	<b>6.057 M</b>	<b>6.945</b>	<b>131</b>	0.356	0.601	0.352	0.710
YOLOV5s	0.805	0.412	6.69 M	15.8	92	0.827	0.839	0.603	0.951
Efficient YOLOV5 (ours)	<b>0.876</b>	0.495	12.97 M	23.3	62	<b>0.880</b>	<b>0.904</b>	<b>0.753</b>	<b>0.967</b>

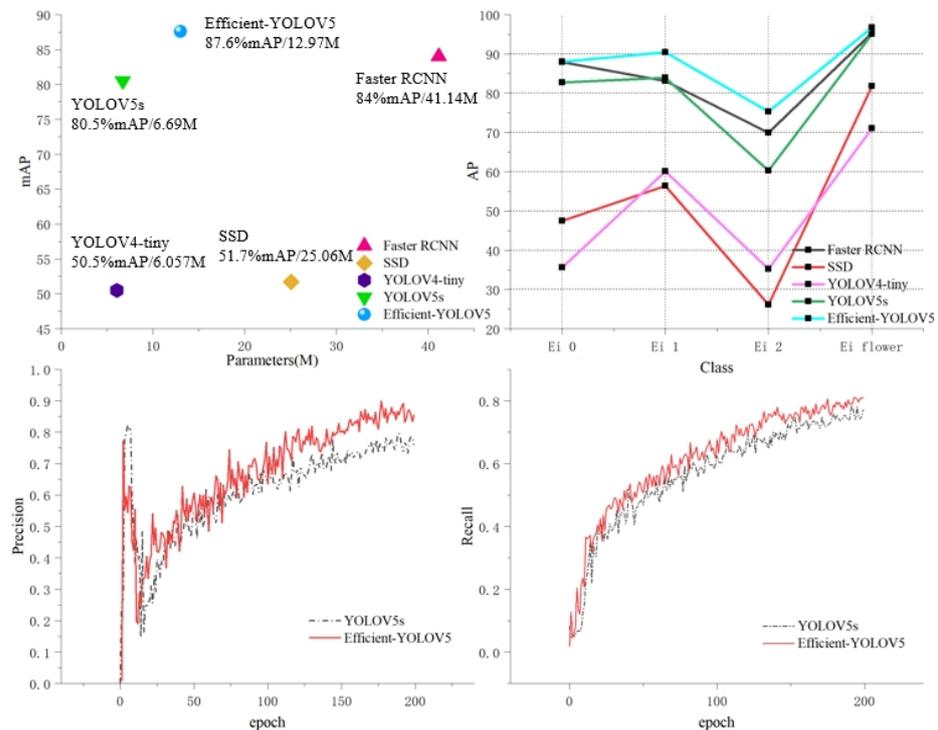


Figure 11. Comparison of efficient YOLOV5 with other algorithms.

The comparison experiment was performed under the framework of YOLOV5, in order to compare the various backbone networks and neck networks, as well as to add different modules

To prove the effectiveness of the design components, in Table 3, the MobileNet [36], GhostNet [37], ShuffleNet [38], PP-LCNet [39], and EfficientNet lightweight network models are selected for feature extraction network comparison. GhostNet uses the Ghost module to achieve light weight by reducing the redundant feature map, and the detection effect of *Eichhornia crassipes* was not good. MobileNetV3Small, ShuffleNetV2, and PP-LCNet all use depthwise separable convolution as basic modules, with good results in the lightweight and FPS values, but with no more than 80% accuracy. Relatively speaking, EffientNet can achieve 87.6% accuracy, while the FPS values are not significantly reduced.

Table 3. Comparison of various feature extraction networks.

Backbone	mAP@50	mAP@.5:.95	Params	GFLOPs	FPS	AP <sub>50</sub> of Classes			
						Ei 0	Ei 1	Ei 2	Ei Flower
MobileNetV3Small	0.724	0.359	12.04 M	20.6	66	0.752	0.735	0.488	0.921
GhostNet	0.793	0.405	14.32 M	22.5	49	0.856	0.848	0.525	0.941
ShuffleNetV2	0.717	0.35	<b>11.86 M</b>	<b>20.4</b>	63	0.756	0.746	0.423	0.941
PP-LCNet	0.676	0.323	12.07 M	20.9	<b>72</b>	0.716	0.707	0.407	0.875
EffientNet-Lite0 (ours)	<b>0.876</b>	<b>0.495</b>	12.97 M	23.3	62	<b>0.880</b>	<b>0.904</b>	<b>0.753</b>	<b>0.967</b>

In Table 4, the influence of adding different modules on the detection performance is compared. The attention mechanisms of CBAM [40], CA [41], ECA [42], and SA mostly adopt the method of combining spatial and channel attention, which is similar in accuracy and has no large gap. Embedding the transformer [43] module with a self-attention mechanism reduces the detection accuracy of *Eichhornia crassipes*. Although the ECA attention mechanism has little increase in the amount of calculation, it has a few effects on improving the performance of the detection network, and the SA module performs better in this regard. In Table 5, FPN, FPN, PAN, and BiFPN [44] were selected for feature network

fusion comparison. The performance of FPN and PAN was 87% mAP, with 60 FPS, and 86.4% BiFPN, with 63 FPS. Both of them have considerable advantages for common FPN feature network fusion. However, there is a bottom-up branch in FPN and PAN, which is more conducive to increasing the RFB module, in order to use multiscale means to increase the performance of the model.

**Table 4.** Comparison of various modules added in efficient YOLOV5.

Module	mAP@50	mAP@.5:.95	Params	GFLOPs	FPS	AP <sub>50</sub> of Classes			
						Ei 0	Ei 1	Ei 2	Ei Flower
CBAM	0.872	0.493	13.05 M	23.3	48	0.861	0.902	<b>0.754</b>	0.971
coordinate attention	0.874	0.494	13.02 M	23.2	50	0.869	0.899	0.749	0.980
Efficient channel attention	0.875	0.497	<b>12.97 M</b>	<b>23.1</b>	54	0.874	0.903	0.753	<b>0.972</b>
Transformer	0.856	0.495	16.35 M	25.6	43	0.867	0.898	0.688	0.972
shuffle attention (ours)	<b>0.876</b>	<b>0.495</b>	12.97 M	23.3	62	<b>0.880</b>	<b>0.904</b>	0.753	0.967

**Table 5.** Comparison of various feature fusion networks.

Neck	mAP@50	mAP@.5:.95	Params	GFLOPs	FPS	AP <sub>50</sub> of Classes			
						Ei 0	Ei 1	Ei 2	Ei Flower
FPN,	0.657	0.285	7.85 M	17.2	71	0.702	0.655	0.451	0.820
FPN, and PAN	0.870	0.492	10.24 M	20.9	60	0.879	<b>0.911</b>	0.732	0.961
BiFPN	0.864	<b>0.496</b>	13.01 M	23.3	63	0.885	0.901	0.701	0.968
FPN, PAN, and RFB (ours)	<b>0.876</b>	0.495	12.97 M	23.3	62	<b>0.880</b>	0.904	<b>0.753</b>	<b>0.967</b>

Figure 12 shows the comparison of the test results of the faster RCNN, SSD, YOLOV4-tiny, YOLOV5s, and efficient YOLOV5 on homemade datasets. The SSD and YOLOV4-tiny algorithms had more missed detections, and the detection effect was poor. In the images with multiple *Eichhornia crassipes* aggregation targets, the faster RCNN and YOLOV5s algorithms also had a certain degree of missed detection. Efficient YOLOV5 could detect all targets with a good recall rate and more accurate frame selection of targets. Figure 13 shows the detection effect of efficient YOLOV5 in different environments, and the detection accuracy of *Eichhornia crassipes* with extremely dense and large coverage was slightly poor. For the main target Ei 1 *Eichhornia crassipes*, the detection was accurate in different complex water environments. In some water areas, there are aquatic plants, floating lotus, and deciduous crops, similar to the shape and color of *Eichhornia crassipes*. At the same time, there was occlusion and overlap of *Eichhornia crassipes* targets. Efficient YOLOV5 can still achieve accurate detection without errors.

### 3.3. Ablation Experiment

To confirm the performance advantages of efficient YOLOV5, based on YOLOV5 with EfficientNet as the backbone, four groups of ablation experiments were designed: 1, without changing activation function; 2, removing SoftPool from SPP; 3, removing the SA attention mechanism; 4, removing the RFB module. To accurately compare the performance of the designed modules, migration learning was not used in the ablation experiments.

From Table 6, it is concluded that the ELU function contributed the most to the whole network performance, and the mAP increased by 4.2%. The ELU function eliminated the problem of neuron death, to some extent, and increased the global modelling ability of the model. The SoftPool and RFB modules had a certain impact on FPS and params, but also increased the mAP value by 0.5% and 0.6%, respectively. From a numerical point of view, adding the SA attention mechanism did not seem to change the network performance

much; however, in the actual detection, as shown in Figure 14, the use of the attention mechanism reduced the missed detection and false detection to a certain extent.

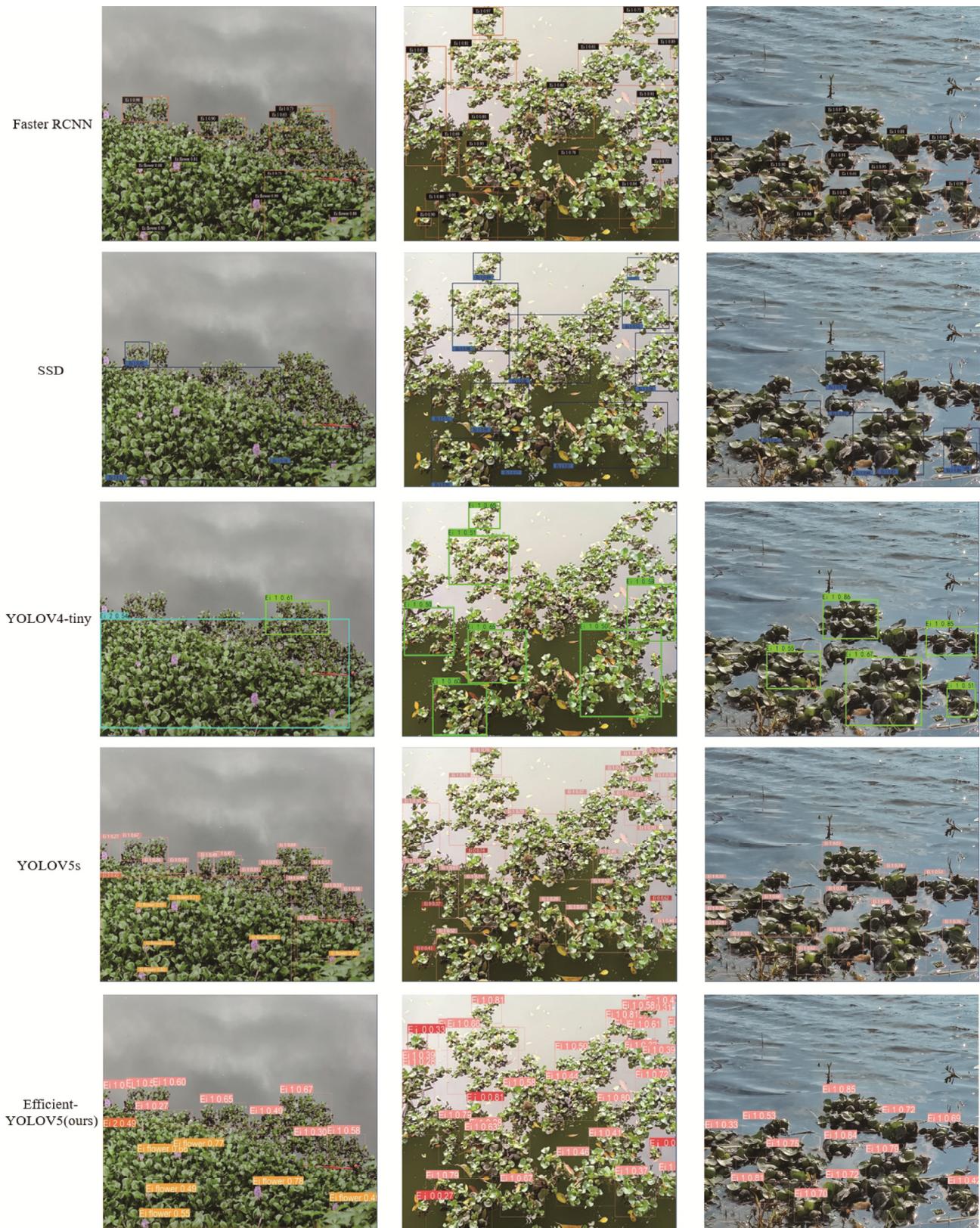


Figure 12. Comparison of detection effects of different algorithms.



Figure 13. Detection effect of efficient YOLOV5 in different environments.

Table 6. Ablation experiment results.

Method	mAP@50	mAP@.5:.95	Params	GFLOPs	FPS	AP <sub>50</sub> of Classes			
						Ei 0	Ei 1	Ei 2	Ei Flower
W/o ELU	0.834	0.445	12.96 M	23.2	63	0.85	0.86	0.734	0.894
W/o SoftPool	0.871	0.491	12.97 M	23.2	68	0.875	0.89	0.752	0.967
W/o SA	0.874	0.495	12.96 M	23.2	62	0.883	0.905	0.746	0.96
W/o RFB	0.87	0.492	10.24 M	20.9	60	0.879	<b>0.911</b>	0.732	0.961
(ours)	<b>0.876</b>	<b>0.495</b>	12.97 M	23.3	62	<b>0.880</b>	0.904	<b>0.753</b>	<b>0.967</b>

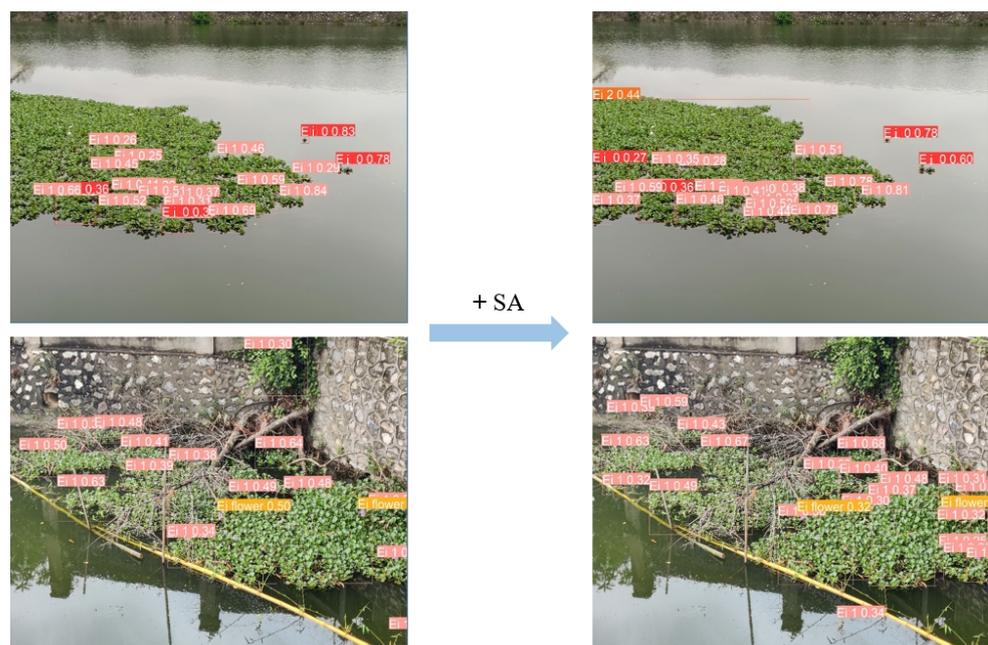


Figure 14. Comparison of detection effects before and after using the SA attention mechanism.

#### 4. Conclusions

In recent years, given the practical application needs of the daily monitoring and cleaning of *Eichhornia crassipes* in small water areas such as streams, small lakes, and ornamental ponds, an unmanned boat was proposed to replace the manual method, and an efficient YOLOV5 was designed for the real-time detection of *Eichhornia crassipes*. Overall, the contributions of this study are as follows. To train the object detection network, images of *Eichhornia crassipes* in different environments, weather, and geographical locations were collected as datasets. For the network model, based on the YOLOV5 model, EfficientNet-Lite was used to replace the backbone network, and the activation function was changed to ELU. The SoftPool and SA modules were embedded into the feature extraction network, and RFB modules were added to the neck network to enhance the multiscale feature fusion ability. Experiments were performed on the working machine. The obtained results show that the aggregation morphology of *Eichhornia crassipes* could be accurately detected in different complex water environments. Compared with the YOLOV5s model, the mAP was increased by 7.1%. For the Ei1 and Ei2 categories with the most overlapping and dense situations, the AP was increased by 6.5% and 15%, respectively. The ablation experiment of efficient YOLOV5 confirmed the effectiveness of the added module. The number of parameters of the model was only 12.97 M, and the FPS was 62 when using the P4000 video card. Under the premise of ensuring the performance, the developed approach can meet the requirement of the real-time and accurate detection of *Eichhornia crassipes* on Jetson nano.

However, the growing water environment of *Eichhornia crassipes* is complex, and the growth aggregation morphology is diverse, which creates certain difficulties in the calibration of datasets and extraction of features in training. The detection accuracy of *Eichhornia crassipes* in some environments, especially in its extremely dense environments, is insufficient. In the future, we should further increase the dataset, enhance the network feature extraction and fusion ability (while ensuring the lightweight nature of the model), and promote the application of the *Eichhornia crassipes* unmanned cleaning boat to more water environments.

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

**Funding:** The work in this paper was supported by the National Key Research and Development Program of China (2021YFD1400100 and 2021YFD1400101) and Guangxi Ba-Gui Scholars Program (2019A33).

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** Not applicable.

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

#### References

1. Chen, X.; Pan, W.; Wang, M. Spatial distribution characteristics and dynamics of *Eichhornia crassipes* in the Shuikou Reservoir, Fujian Province. *J. Lake Sci.* **2012**, *24*, 391–399.
2. Wang, Z.; Zhu, P.; Sheng, P.; Zheng, J. Biological characteristics of water hyacinth. *Jiangsu J. Agric. Sci.* **2011**, *27*, 531–536.
3. Merry, M.; Mitan, N. Water hyacinth: Potential and Threat. *Mater. Today Proc.* **2019**, *19*, 1408–1412. [[CrossRef](#)]
4. Hill, J.M.; Hutton, B.; Steffins, K.; Rieucan, G. Floating along marsh edges: The impact of invasive water hyacinth (*Eichhornia crassipes*) on estuarine species assemblages and predation risk. *J. Exp. Mar. Biol. Ecol.* **2021**, *544*, 151618. [[CrossRef](#)]
5. Gao, L.; Li, B. The study of a specious invasive plant water hyacinth (*Eichornia crassipes*): Achievements and challenges. *Acta Phytoecol. Sin.* **2004**, *28*, 735–752.

6. Chu, J.J.; Ding, Y.; Zhuang, Q.J. Invasion and control of water hyacinth (*Eichhornia crassipes*) in China. *J. Zhejiang Univ. Sci. B* **2006**, *7*, 623–626. [[CrossRef](#)]
7. Hidayati, N.; Soeprbowati, T.R.; Helmi, M. The evaluation of water hyacinth (*Eichhornia crassiper*) control program in Rawapening Lake, Central Java Indonesia. *IOP Conf. Series. Earth Environ. Sci.* **2018**, *142*, 12016. [[CrossRef](#)]
8. Zhang, L.; Zhu, P.; Gao, Y.; Zhang, Z.; Yan, S. Design of anti-stormy wave enclosures for confined growth of water hyacinth in lakes. *Jiangsu J. Agric. Sci.* **2013**, *29*, 1360–1364.
9. Zhang, Z.; Qin, H.; Liu, H.; Li, X.; Wen, X.; Zhang, Y.; Yan, S. Effect of Large-Scale Confined Growth of Water Hyacinth Improving Water Quality of Relatively Enclosed Eutrophicated Waters in Caohai of Lake Dianchi. *J. Ecol. Rural. Environ.* **2014**, *30*, 306–310.
10. Zheng, B.; Lu, J. Inhibitory effects of harlequin glory-bower (*Clerodendrum trichotomum*) extract on growth of water hyacinth (*Eichhornia crassiper*). *J. Zhejiang Univ. (Agric. Life Sci.)* **2012**, *38*, 279–287. [[CrossRef](#)]
11. Yan, S.H.; Song, W.; Guo, J.Y. Advances in management and utilization of invasive water hyacinth (*Eichhornia crassipes*) in aquatic ecosystems—a review. *Crit. Rev. Biotechnol.* **2017**, *37*, 218–228. [[CrossRef](#)]
12. Sun, L.; Zhu, Z.S. An Area Growth Model of *Eichhornia Crassipes* with Application to Lake Ecosystem Restoration. *Appl. Mech. Mater.* **2014**, *496*, 3009–3012. [[CrossRef](#)]
13. Zhou, Z.; Li, J.; Wang, Y.; Qiu, J.; Zhang, X.; Zu, C.; Guo, M. Free Growth and Diffusion of Water Hyacinth Based on Logistic-CA and Differential Equations. In Proceedings of the CSAE 2020: 4th International Conference on Computer Science and Application Engineering, Sanya, China, 20–22 October 2020. [[CrossRef](#)]
14. Mukarugwiro, J.A.; Newete, S.W.; Adam, E.; Nsanganwimana, F.; Abutaleb, K.; Byrne, M.J. Mapping spatio-temporal variations in water hyacinth (*Eichhornia crassipes*) coverage on Rwandan water bodies using multispectral imageries. *Int. J. Environ. Sci. Technol.* **2021**, *18*, 275–286. [[CrossRef](#)]
15. Worqlul, A.W.; Ayana, E.K.; Dile, Y.T.; Moges, M.A.; Dersseh, M.G.; Tegegne, G.; Kibret, S. Spatiotemporal Dynamics and Environmental Controlling Factors of the Lake Tana Water Hyacinth in Ethiopia. *Remote Sens.* **2020**, *12*, 2706. [[CrossRef](#)]
16. Sun, L.; Zhu, Z. Modelling yield of water hyacinth (*Eichhornia crassipes*) using satellite and GPS sensors. In Proceedings of the 2017 6th International Conference on Agro-Geoinformatics, Fairfax, VA, USA, 7–10 August 2017. [[CrossRef](#)]
17. Dube, T.; Mutanga, O.; Sibanda, M.; Bangamwabo, V.; Shoko, C. Testing the detection and discrimination potential of the new Landsat 8 satellite data on the challenging water hyacinth (*Eichhornia crassipes*) in freshwater ecosystems. *Appl. Geogr.* **2017**, *84*, 11–22. [[CrossRef](#)]
18. Thamaga, K.H.; Dube, T. Testing two methods for mapping water hyacinth (*Eichhornia crassipes*) in the Greater Letaba river system, South Africa: Discrimination and mapping potential of the polar-orbiting Sentinel-2 MSI and Landsat 8 OLI sensors. *Int. J. Remote. Sens.* **2018**, *39*, 8041–8059. [[CrossRef](#)]
19. Thamaga, K.H.; Dube, T. Remote sensing of invasive water hyacinth (*Eichhornia crassipes*): A review on applications and challenges. *Remote Sens. Appl. Soc. Environ.* **2018**, *10*, 36–46. [[CrossRef](#)]
20. Feng, Z.; Pan, F.; Li, Y. Image recognition based on water hyacinth controlled breeding monitoring equipment. *J. Phys. Conf. Ser.* **2020**, *1549*, 32116. [[CrossRef](#)]
21. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 1137–1149. [[CrossRef](#)] [[PubMed](#)]
22. 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. [[CrossRef](#)]
23. Redmon, J.; Farhadi, A. YOLOv3: An Incremental Improvement. *arXiv* **2018**, arXiv:1804.02767.
24. Bochkovskiy, A.; Wang, C.; Liao, H. YOLOv4: Optimal Speed and Accuracy of Object Detection. *arXiv* **2020**, arXiv:2004.10934.
25. Du, L.; Chen, W.; Fu, S.; Kong, H.; Li, C.; Pei, Z. Real-time Detection of Vehicle and Traffic Light for Intelligent and Connected Vehicles Based on YOLOv3 Network. In Proceedings of the 5th International Conference on Transportation Information and Safety (ICTIS), Liverpool, UK, 14–17 July 2019.
26. Kim, J. Vehicle Detection Using Deep Learning Technique in Tunnel Road Environments. *Symmetry* **2020**, *12*, 2012. [[CrossRef](#)]
27. Hu, X.; Liu, Y.; Zhao, Z.; Liu, J.; Yang, X.; Sun, C.; Chen, S.; Li, B.; Zhou, C. Real-time detection of uneaten feed pellets in underwater images for aquaculture using an improved YOLO-V4 network. *Comput. Electron. Agric.* **2021**, *185*, 106135. [[CrossRef](#)]
28. Zhao, T.; Yi, X.; Zeng, Z.; Feng, T. MobileNet-Yolo based wildlife detection model: A case study in Yunnan Tongbiguan Nature Reserve, China. *J. Intell. Fuzzy Syst.* **2021**, *41*, 2171–2181. [[CrossRef](#)]
29. Ge, Y.; Lin, S.; Zhang, Y.; Li, Z.; Cheng, H.; Dong, J.; Shao, S.; Zhang, J.; Qi, X.; Wu, Z. Tracking and Counting of Tomato at Different Growth Period Using an Improving YOLO-Deepsort Network for Inspection Robot. *Machines* **2022**, *10*, 489. [[CrossRef](#)]
30. Wang, X.; Zhao, Q.; Jiang, P.; Zheng, Y.; Yuan, L.; Yuan, P. LDS-YOLO: A lightweight small object detection method for dead trees from shelter forest. *Comput. Electron. Agric.* **2022**, *198*, 107035. [[CrossRef](#)]
31. Tan, M.; Le, Q.V. EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks. In Proceedings of the 36th International Conference on Machine Learning (ICML), Long Beach, CA, USA, 9–15 June 2019.
32. Clevert, D.; Unterthiner, T.; Hochreiter, S. Fast and Accurate Deep Network Learning by Exponential Linear Units(ELUs). *arXiv* **2016**, arXiv:1511.07289.
33. Stergiou, A.; Poppe, R.; Kalliatakis, G. Refining activation downsampling with SoftPool. *arXiv* **2021**, arXiv:2101.00440v1.

34. Zhang, Q.; Yang, Y. Shuffle Attention for Deep Convolutional Neural Networks. In Proceedings of the ICASSP 2021–2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Toronto, ON, Canada, 13 May 2021; pp. 2235–2239. [[CrossRef](#)]
35. Liu, S.; Huang, D.; Wang, Y. Receptive Field Block Net for Accurate and Fast Object Detection. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 385–400.
36. Howard, A.; Sandler, M.; Chen, B.; Wang, W.; Chen, L.-C.; Tan, M.; Chu, G.; Vasudevan, V.; Zhu, Y.; Pang, R.; et al. Searching for MobileNetV3. In Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Korea, 27 October–2 November 2019; pp. 1314–1324.
37. Han, K.; Wang, Y.; Tian, Q.; Guo, J.; Xu, C.; Xu, C. GhostNet: More features from cheap operations. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 14–19 June 2020; pp. 1580–1589.
38. Ma, N.; Zhang, X.; Zheng, H.-T.; Sun, J. ShuffleNet V2: Practical Guidelines for Efficient CNN Architecture Design. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 116–131. [[CrossRef](#)]
39. Cui, C.; Gao, T.; Wei, S.; Du, Y.; Guo, R.; Dong, S.; Lu, B.; Zhou, Y.; Lv, X.; Liu, Q.; et al. PP-LCNet: A Lightweight CPU Convolutional Neural Network. *arXiv* **2021**, arXiv:2109.15099.
40. Woo, S.; Park, J.; Lee, J.Y.; Kweon, I.S. CBAM: Convolutional block attention module. *arXiv* **2018**, arXiv:1807.06521.
41. Hou, Q.; Zhou, D.; Feng, J. Coordinate Attention for Efficient Mobile Network Design. In Proceedings of the 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Nashville, TN, USA, 20–25 June 2021.
42. Wang, Q.; Wu, B.; Zhu, P.; Li, P.; Zuo, W.; Hu, Q. ECA-Net: Efficient Channel Attention for Deep Convolutional Neural Networks. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 14–19 June 2020; pp. 11531–11539.
43. Vaswani, A.; Shazer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, L.; Polosukhin, I. Attention Is All You Need. *arXiv* **2017**, arXiv:1706.03762.
44. Tan, M.; Pang, R.; Le, Q.V. EfficientDet: Scalable and efficient object detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 10778–10787.