Article

CURI-YOLOv7: A Lightweight YOLOv7tiny Target Detector for Citrus Trees from UAV Remote Sensing Imagery Based on Embedded Device

Yali Zhang [1,2,3], Xipeng Fang [1,3], Jun Guo [1,3], Linlin Wang [4], Haoxin Tian [5], Kangting Yan [3,6] and Yubin Lan [2,3,6,*]

1 College of Engineering, South China Agricultural University, Wushan Road, Guangzhou 510642, China; ylzhang@scau.edu.cn (Y.Z.); xpfang@stu.scau.edu.cn (X.F.); junguo@stu.scau.edu.cn (J.G.)
2 Guangdong Laboratory for Lingnan Modern Agriculture, Guangzhou 510642, China
3 National Center for International Collaboration Research on Precision Agricultural Aviation Pesticide Spraying Technology, Guangzhou 510642, China; ktyan@stu.scau.edu.cn
4 School of Artificial Intelligence, Shenzhen Polytechnic University, Shenzhen 518055, China; wanglinlin@szpt.edu.cn
5 College of Automation Engineering, Nanjing University of Aeronautics and Astronautics, Nanjing 211106, China; hxtian@stu.scau.edu.cn
6 College of Electronic Engineering and College of Artificial Intelligence, South China Agricultural University, Wushan Road, Guangzhou 510642, China
* Correspondence: ylan@scau.edu.cn

Abstract: Data processing of low-altitude remote sensing visible images from UAVs is one of the hot research topics in precision agriculture aviation. In order to solve the problems of large model size with slow detection speed that lead to the inability to process images in real time, this paper proposes a lightweight target detector CURI-YOLOv7 based on YOLOv7tiny which is suitable for individual citrus tree detection from UAV remote sensing imagery. This paper augmented the dataset with morphological changes and Mosica with Mixup. A backbone based on depthwise separable convolution and the MobileOne-block module was designed to replace the backbone of YOLOv7tiny. SPPF (spatial pyramid pooling fast) was used to replace the original spatial pyramid pooling structure. Additionally, we redesigned the neck by adding GSConv and depth-separable convolution and deleted its input layer from the backbone with a size of (80, 80) and its output layer from the head with a size of (80, 80). A new ELAN structure was designed, and the redundant convolutional layers were deleted. The experimental results show that the GFLOPs = 1.976, the parameters = 1.018 M, the weights = 3.98 MB, and the mAP = 90.34% for CURI-YOLOv7 in the UAV remote sensing imagery of the citrus trees dataset. The detection speed of a single image is 128.83 on computer and 27.01 on embedded devices. Therefore, the CURI-YOLOv7 model can basically achieve the function of individual tree detection in UAV remote sensing imagery on embedded devices. This forms a foundation for the subsequent UAV real-time identification of the citrus tree with its geographic coordinates positioning, which is conducive to the study of precise agricultural management of citrus orchards.

Keywords: citrus trees; remote sensing; YOLOv7; lightweight; target detector

1. Introduction

Agricultural aviation constitutes a pivotal component of modern agriculture [1]. In recent years, there has been a notable surge in the development of agricultural aircrafts in China [2]. In tandem with the inception and advancement of precision agricultural aviation [3], the utilization of UAV remote sensing has the potential to harness fine-scale and dynamically continuous monitoring across farmlands [4]. This technology harbors extensive prospective applications in monitoring crop growth trends [5]. Notably, China stands as one of the foremost citrus-producing nations [6]. Leveraging UAV imagery

presents inherent advantages in fruit tree management, encompassing tasks such as quantification [7], surveillance [8], pest and disease monitoring [9], prescription mapping [10], and identification as well as localization [11]. In the realm of research, endeavors have been undertaken to amalgamate deep learning with UAV imagery [12], facilitating information extraction [13] and segmentation [14]. This amalgamation furnishes a potent instrument for the evolution of precision agriculture, aimed at enhancing productivity and optimizing resource allocation.

An increasing number of scholars are refining existing deep learning algorithms to meet the demands of UAV image detection. Wang et al. [15] synergized YOLOX with migration learning to enhance the accuracy of male cob detection within UAV images, thereby furnishing a high-precision detection methodology tailored for UAV imagery of male corn cobs. Bao et al. [16] introduced a wheat count detection model using TPH-YOLO (YOLO with transformer prediction heads). They employed transformer prediction heads with transfer learning and other strategies to elevate wheat counting accuracy in UAV images. Zhang et al. [17] adopted data augmentation through Mixup and replaced the C3 module with the GhostBottleneck module. Coupled with the Shuffle Attention (SA) module, this approach heightened the focus on small targets. Their creation, YOLOv5s Pole, is geared towards the deployment of agricultural UAV airborne equipment. Luo et al. [18] put forth a YOLO-DRONE that employs distinct activation functions in shallow and deep networks. This algorithm simplifies the recognition of UAV remote sensing images. Zhu et al. [19] introduced the YOLOv4 -Mobilenetv3—CBAM—ASFF—P algorithm, which significantly reduces the model's size compared to the original YOLOv4 while achieving a 98.21% mAP. This adaptation enhances the accuracy of detecting canopy layers in UAV photographs of fruit trees.

The algorithms advanced by scholars in the aforementioned study indeed exhibited enhanced detection accuracy in comparison to the original algorithms. However, these algorithms suffer from challenges such as high GFLOPs and parameter counts, as well as weighty files, resulting in sluggish performance on embedded devices. Consequently, they are unsuitable for processing UAV aerial images targeting citrus trees. Thus, it is of great significance to develop a target detector that can be carried on an embedded device and achieve the imagery processing of citrus orchards. Due to the limited arithmetic power of the embedded devices carried by agricultural UAVs [20,21], this study concentrates on a specific application scenario involving UAV aerial images of citrus trees. The primary emphasis is on strategies to curtail the GFLOPs requirements of the airborne model, optimize the balance between the number of parameters and the model size, and enhance the model's detection speed when operating on embedded devices.

This study introduces a lightweight YOLOv7tiny target detector named CURI-YOLOv7, specifically designed for detecting citrus trees using UAV remote sensing imagery on embedded devices. To achieve this, several key designs choices have been made. First, the CURI-YOLOv7 utilizes a MobileOne-block based backbone structure, leading to a substantial reduction in the model's parameter count. Second, an SPPF structure is employed in place of the original structure, significantly enhancing the model's inference speed. Third, a lightweight ELAN (L-E) structure is adopted as a replacement for ELAN, thereby mitigating the impact of non-critical information on the model. Finally, the neck structure is revamped using GSConv and depth-separable convolution techniques, effectively eliminating the cumbersome target detection layer and thereby resolving the issue of sluggish inference speed on embedded devices. In comparison to alternative target detection algorithms, CURI-YOLOv7 boasts a more lightweight overall network structure. Simultaneously, there are reductions in both GFLOPs and parameter counts, accompanied by a notable improvement in frames per second (FPS). This combination of enhancements renders CURI-YOLOv7 particularly well-suited for deployment on agricultural UAVs and associated image processing tasks. Furthermore, it aligns seamlessly with the application scenario of UAV aerial citrus tree detection, making it an appropriate choice for precision management of citrus orchards.

2. Materials and Methods

2.1. Collection and Production of Datasets

The citrus tree images were collected at citrus orchards in Sihui City, Guangdong Province, China (23°36′N, 112°68′E), as shown in Figure 1. The temperature at the time of the experiment was 35.2 °C, and the humidity was 61.7%. A DJI Phantom 4 RTK quadrotor UAV was selected to collect data. The flight altitude of the aircraft was set to 50 m, the heading overlap rate was 80%, the side overlap rate was 80%, the camera angle was 90°, the ground sample distance (GSD) was 1.3699 cm/pixel, and the pixel size was 4864 × 3648 (pixels). After data augmentation, 1640 images were obtained for this experiment. The topography of the citrus groves is close to the plain, with an average slope of less than 5°. The citrus trees inside the zones were 1.5–2 m tall with an average crown diameter of 2 m. The trees were all spaced about 3 m apart. Due to the presence of water tanks and irrigation pipes throughout the rows of fruit trees, it was not possible for ground equipment to pass between the rows of citrus trees. Therefore, the dataset could only be collected using aerial photography from UAVs and could not be collected by ground trolleys or other ground equipment.
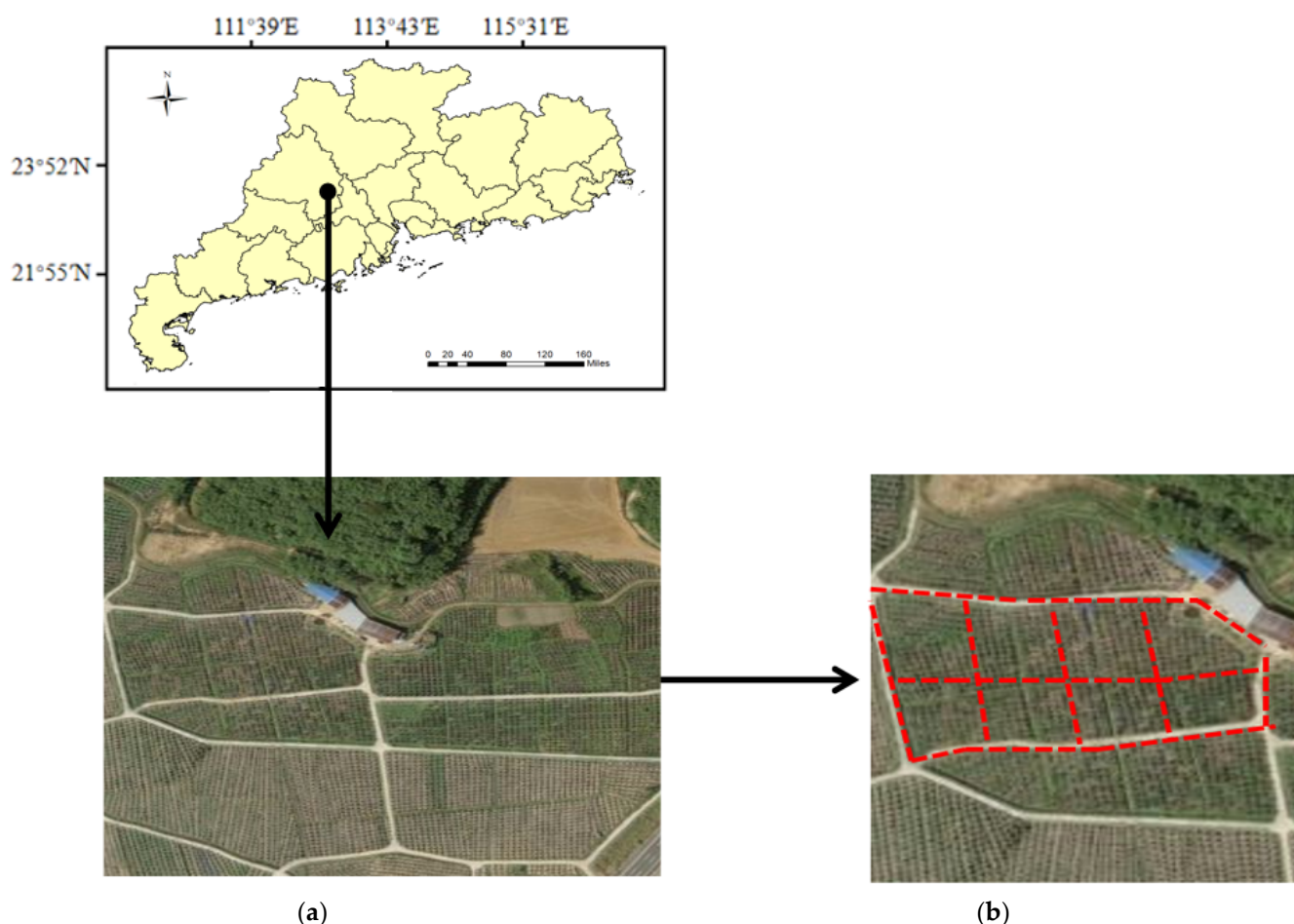

(a)

(b)

Figure 1. Location of data collection sites (23°36′N, 112°68′E): (**a**) shows the experiment area located in Guangdong; (**b**) the dashed box shows part of the collection area.

CNNs often face the risk of overfitting caused by limited data [22]. To address this problem, this study used various data augmentation methods, including morphological processing data augmentation, Mosica data augmentation, and Mixup data augmentation. Morphological processing includes rotation and overturn. It can improve the robustness of the model and improve models' performance. Mosica and Mixup data augmentation are the original data augmentation methods included in the YOLOv7 target detector. Mosica data

augmentation can effectively improve the model's ability to learn the correlation between objects in the dataset. Mixup data augmentation can effectively reduce the overfitting phenomenon and improve the generalization ability of the model. Figure 2 shows a demonstration of morphological processing data augmentation.
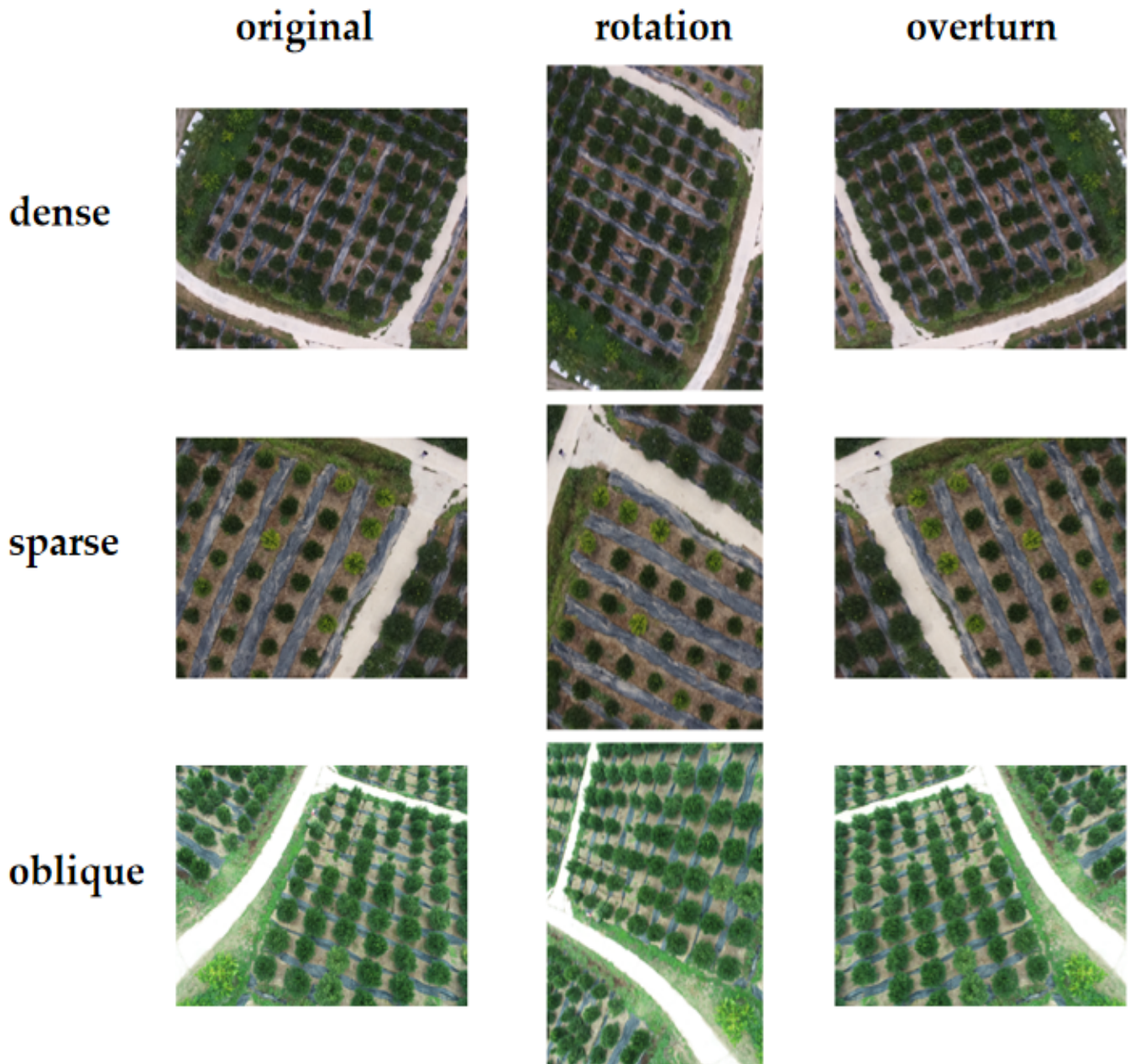
Figure 2. Example of morphological data augmentation.

Manual annotation of images is also required in order to obtain accurate data parameters [23]. In this study, labelimg [24] was used as the dataset labelling tool. Rectangular labels were used for labelling and saved as XML files in Pascal VOC format. A total of 1640 images were divided into (training set:validation set):test set = (9:1):1. At the time of labelling, citrus trees were not labelled if they were obscured by more than two-thirds in the corners of the image. Similarly, citrus trees in the seedling stage or smaller-sized citrus trees were not labeled.

2.2. CURI-YOLOv7 Structure Detail

CURI-YOLOv7 is a weight-improved target detector based on YOLOv7tiny. It focuses on enhancing the detection speed on both computer and embedded devices while minimizing any potential degradation in target detection precision and recall. Additionally, CURI-YOLOv7 effectively reduces the number of parameters and GFLOPs required by the model. The network structure diagram is shown in Figure 3, and the main improvements reducing its weight are as follows:

(1)   Redesigned backbone based on MobileOne-block structure
(2)   Replaced the original structure with a faster structure named SPPF
(3)   Removed the big target detection layer and redesigned the input structure and output structure of PANet
(4)   Replaced the ELAN structure in YOLOv7tiny with Lightweight-ELAN (L-E).
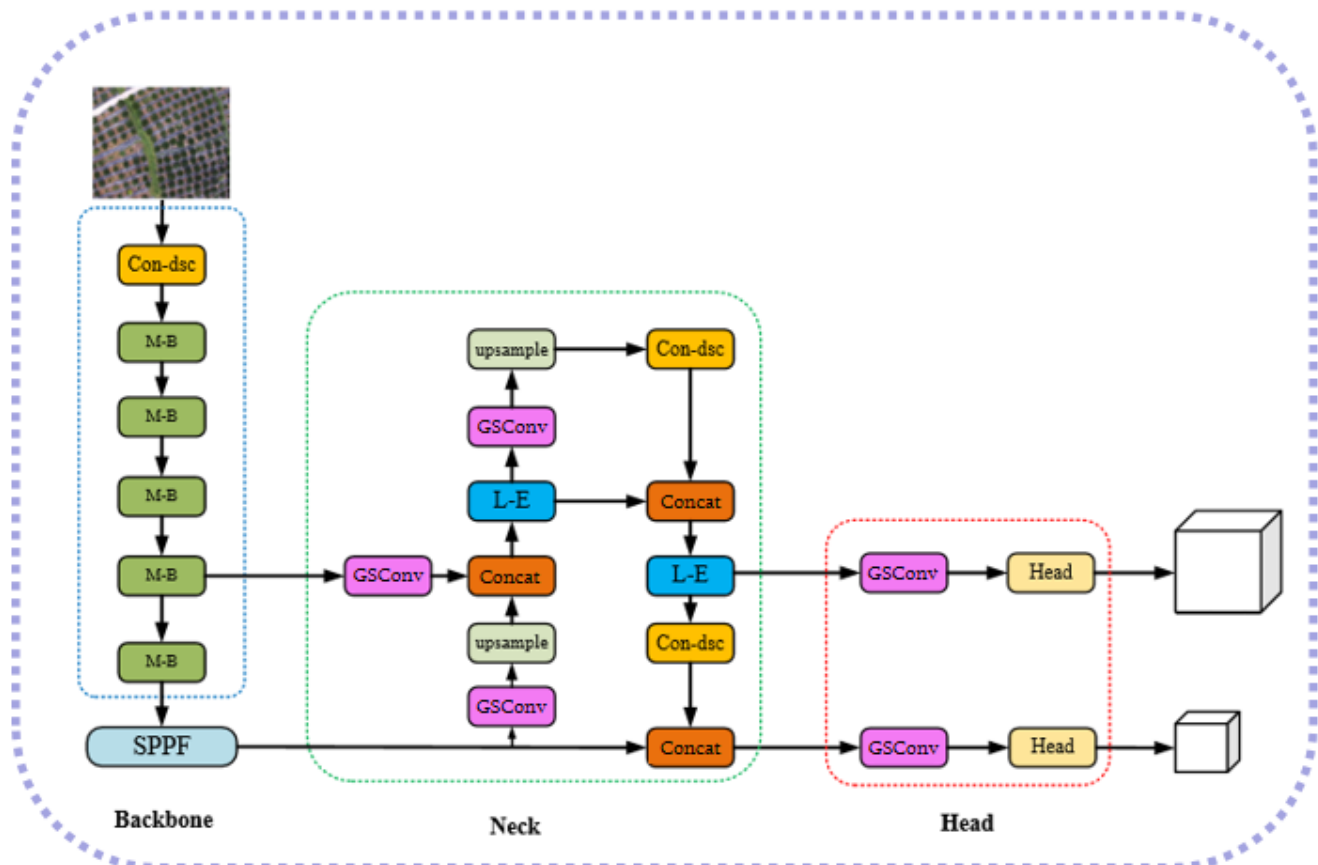(5)   Replaced the convolutional layer in the neck network using GSConv and depthwise separable convolution.

Figure 3. Structure of CURI-YOLOv7.

2.2.1. Construction of Backbone

MobileOne [25] is a lightweight backbone developed by Vasu. et al. in 2022 that features a simple and lightweight architecture. The structure of MobileOne is displayed in Figure 4. In terms of the concept of reparameterization, MobileOne incorporates a multi-branch structure that significantly reduces parameters and enhances the computational speed of the model.

The CURI-YOLOv7 target detector refers to MobileOne, utilizing the MobileOne-block as the base module of the backbone. Its structure and the number of input and output channels are shown in Table 1. This paper exclusively employs the training structure of MobileOne-block. Con_des in the table represents the depthwise separable convolution,

and the input shape of CURI-YOLOv7 is (640, 640, 3). To adjust the input channels of the backbone, a depthwise separable convolution is added. The utilization of the designed CURI-YOLOv7′s backbone instead of YOLOv7′s CSPDARKNET indeed leads to an improvement in the inference speed of the model. This enables the deployment of inference models in embedded devices and the real-time target detection of individual trees in citrus orchards in our study.
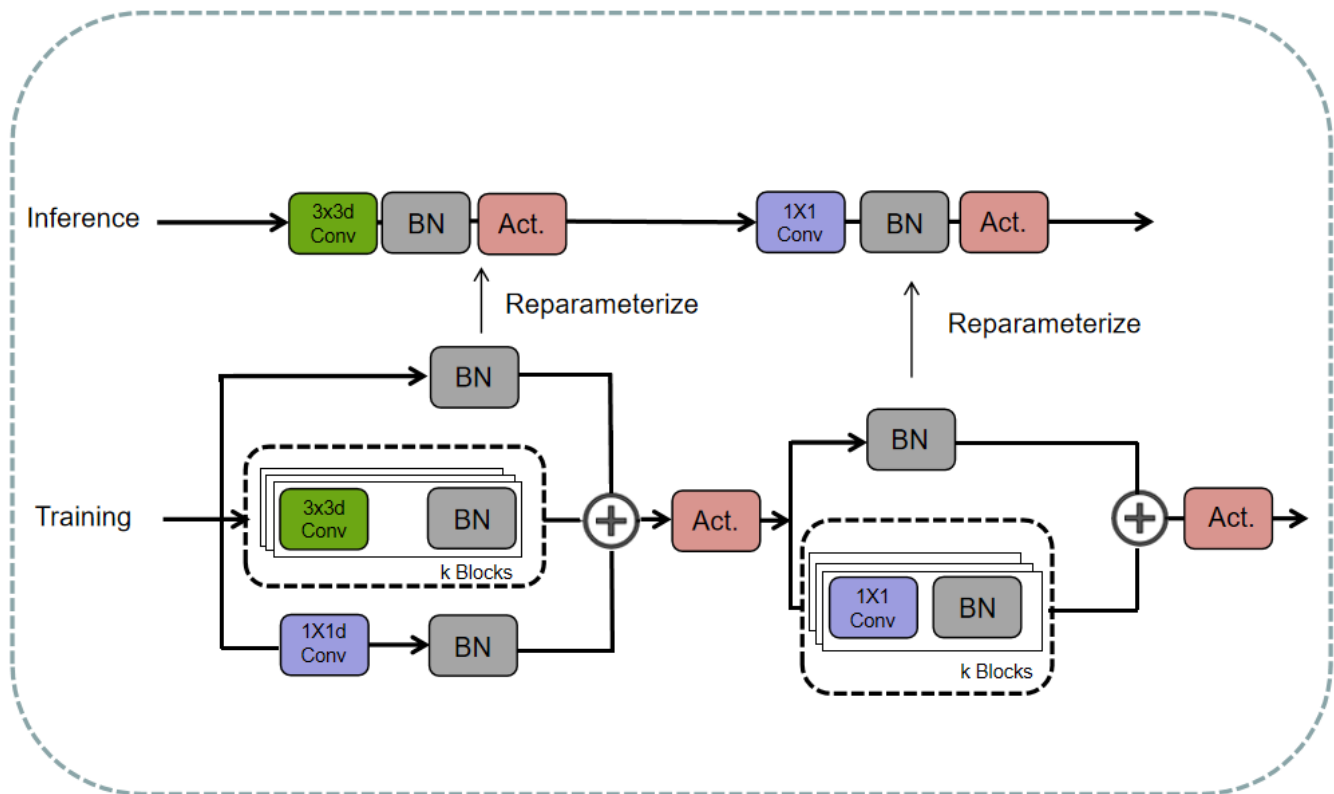

Figure 4. Structure of MobileOne-block.

Table 1. Structure and parameters of the backbone for CURI-YOLOv7.

Stage	Block-Type	Stride	Kernel Size	Input Channels	Output Channels
1	Con des	1		3	16
2	MobileOne-block	2	1	16	32
3	MobileOne-block	2	1	32	64
4	MobileOne-block	2	1	64	128
5	MobileOne-block	2	1	128	256
6	MobileOne-block	2	1	256	512

2.2.2. Spatial Pyramid Pooling and ELAN Improvements

The spatial pyramid pooling used in YOLOv7 is the SPPCSPC structure, which shows excellent performance, but the number of parameters and GFLOPs computed are substantially higher. Given that this study focuses exclusively on recognizing citrus trees as the target, it is worth noting that the structure and morphology of citrus trees are relatively simple. Consequently, it is possible to opt for a spatial pyramid pooling that may result in a slight decrease in detection accuracy compared to SPPCSPC. However, the advantage lies in achieving a notable reduction in both the number of parameters and GFLOPs. Jocher et al. adopted an improved spatial pyramid pooling fast (SPPF) [26] structure to replace the original SPP. The SPPF achieves the same computation result as the original parallel MaxPool layers of three different sizes by serializing multiple MaxPool layers of the same

size, greatly reducing the computation time [27]. The main difference between the two is that the speed of inference varies greatly. The structures of SPPCSPC and SPPF are shown in Figure 5.
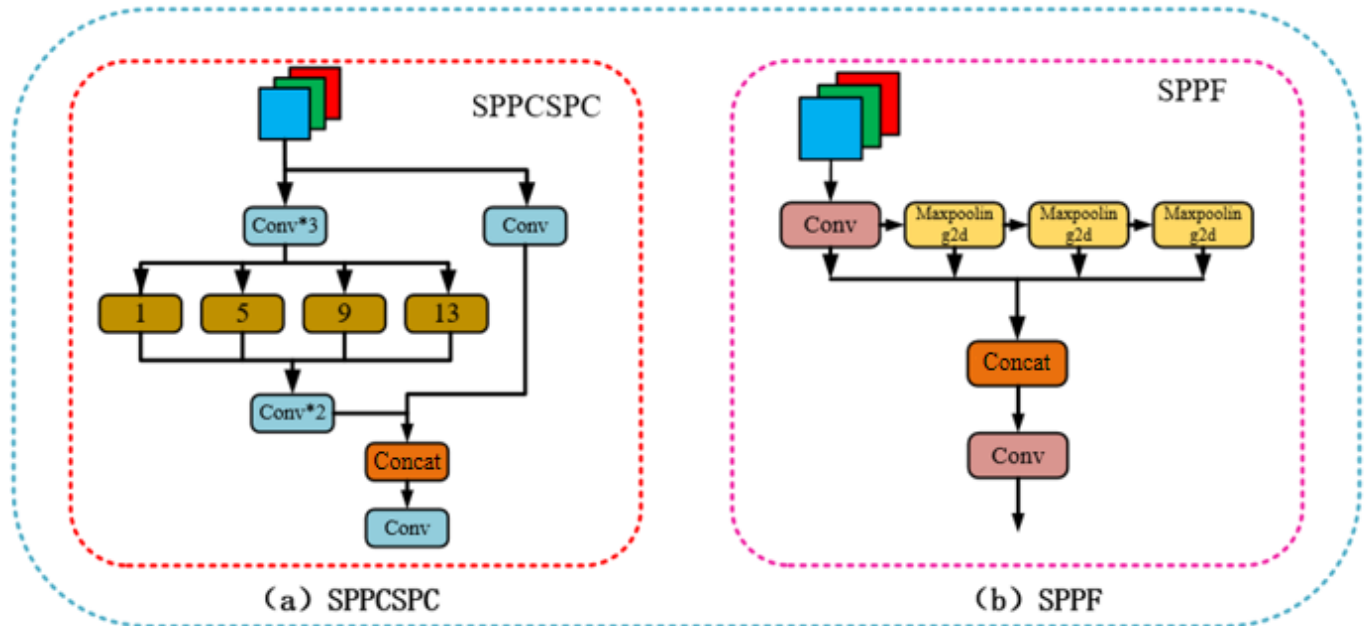

Figure 5. Comparison between the structures of SPPCSPC and SPPF: (**a**) shows the SPPCSPC structure, and (**b**) shows the SPPF structure.

ELAN (extended efficient layer aggregation networks), designed by Wang et al. [28], were first used in the YOLOv7 target detector. Considering the requirements in terms of inference speed in embedded devices, this study designs a lightweight ELAN structure: L-ELAN (lightweight-extended efficient layer aggregation networks). By modifying the paths of ELAN, convolutional layers that are not useful for our target detector are deleted. However, the design concept of YOLOv7 and the fundamental design concept of the ELAN structure remain unchanged. By performing tests and identifying convolutional layers within the ELAN structure that are not useful for this research dataset, these identified convolutional layers were deleted, and the L-ELAN structure was created. Specifically, the focus was on identifying convolutional layers that have little impact on the mean average precision (MAP) after their removal. Figure 6 provides a visual comparison between ELAN in YOLOv7tiny and L-ELAN.

2.2.3. Construction of the Input and Output Layers of the Neck

In both YOLOv7 and YOLOv7tiny, the output feature layer sizes are (80, 80), (40, 40), and (20, 20), respectively. The corresponding detection target types are small targets, medium targets, and large targets [29]. Since convolutional kernels of different sizes can obtain information on different sizes of imagery, when extracting features for small targets, the effect of retaining more local feature information on small targets can be achieved by selecting the appropriate perceptual field size or considering multi-scale perceptual fields [30].

In the dataset constructed for our study, the citrus trees in the images are usually similar and fixed in size because the height of the aerial photograph is set to 50 m. YOLOv7 has a wide feature receptive field, but in our study, it was not necessary to use all of the output feature layers in order to obtain a wide receptive field. To reduce the number of parameters and GFLOPs and improve the inference speed on the embedded device, the proposed CURI-YOLOv7 target detector deletes the smaller target detection layer with output feature layer size (80, 80), deletes the feature layer with input size (80, 80) from the

neck network, and redesigns the neck network to match the number of input and output channels on the neck of YOLOv7tiny.
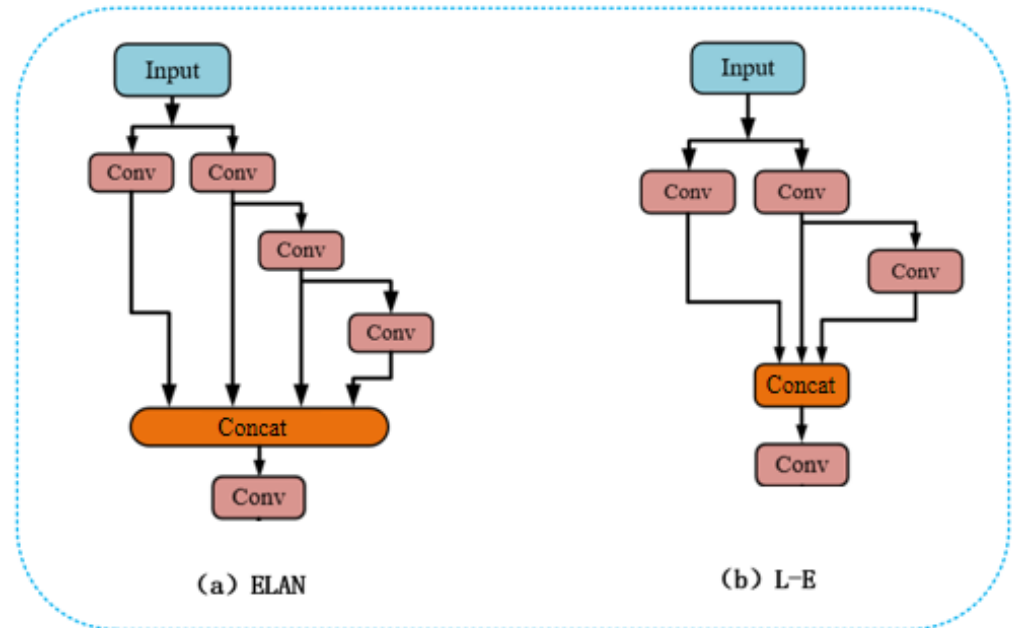

Figure 6. Comparison between ELAN in YOLOv7tiny and L-E.

2.2.4. Fusion of GSConv and Depthwise Separable Convolution into the Neck

CURI-YOLOv7 adopts GSConv [31] and depthwise separable convolution [32] instead of normal convolution in order to reduce the parameters and inference speed, and its structure diagram is shown in Figure 7. Through extensive experiments, we found the best alternative solutions for GSConv and depthwise separable convolution within the design framework of CURI-YOLOv7.
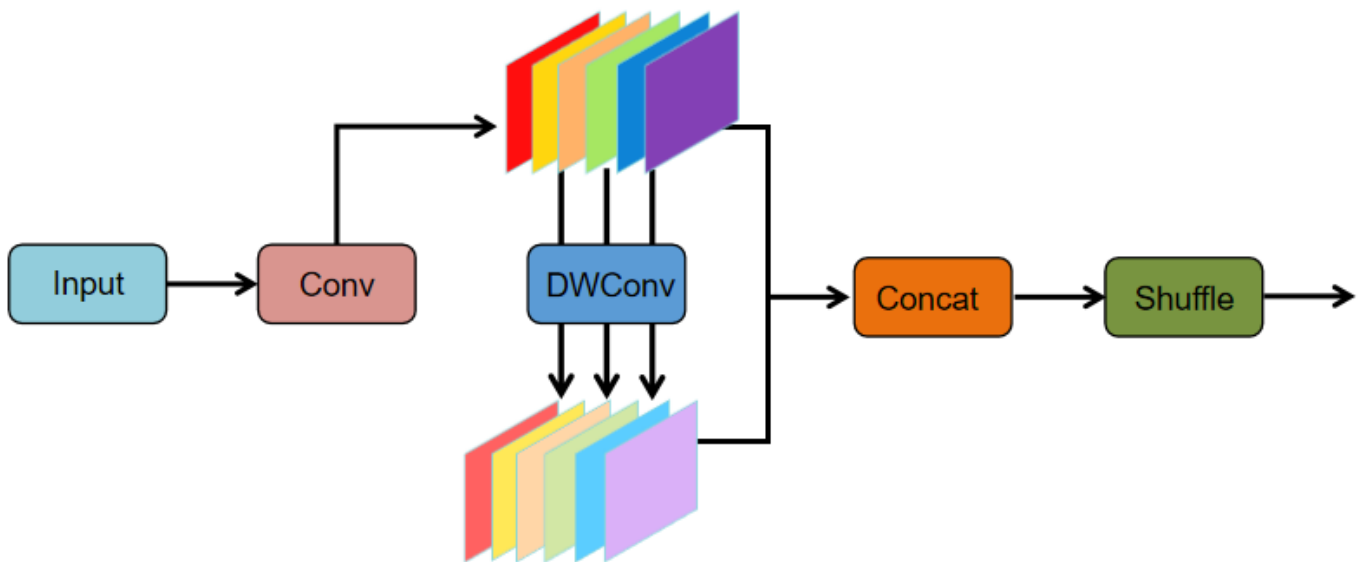

Figure 7. The structure diagram of GSConv.

GSConv is a variant of a convolutional neural network. GSConv retains as many of these connections as possible while keeping the time complexity small, which reduces information loss and enables faster operation [33].

Depthwise separable convolution (DSC) [32] is a convolutional operation whose structure is depicted in Figure 8. The calculation process is described by Equation (1). G stands for the output feature; K stands for the convolution kernel; F stands for the input feature; i,j is the feature pixel position; k,l is the output feature resolution; and m is the number of channels [34]. DSC has gained widespread adoption in recent years for model lightweighting studies. In comparison to standard convolution, it achieves the same feature extraction effect. The key advantage of this convolution is its effective reduction of model parameters.

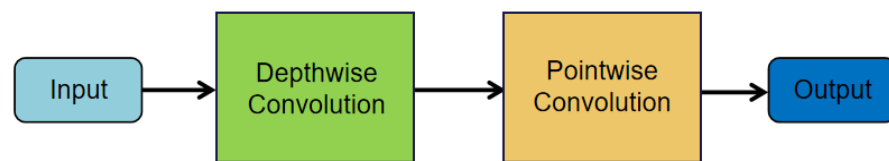$$G_{k,l,m} = \sum_{i,j} K_{i,j,m} \cdot F_{K+i-1,l+j-1,m} \tag{1}$$

Figure 8. The structure diagram of depthwise separable convolution.

2.3. Embedded Device

The embedded device Jetson Xavier Nx (NVIDIA's GPU edge-computing device to be released in 2020 [35]) with high performance [36] was chosen for our experiment. The Jetson Xavier Nx was equipped with the Jetson Xavier Nx Developer Kit with Ubuntu 18.04, and the Python 3.6 and PyTorch environments were configured for this study. The details of the embedded devices are shown in Table 2.

Table 2. Technical specifications of the embedded devices.

CPU	GPU	Size	DLA	Vision Accelerator
6-core NVIDIA Carmel Arm v8.2 64-bit CPU	384-core NVIDIA Volta™ architecture GPU with 48 Tensor Cores	69.6 mm × 45 mm	2 × NVDLA	2 × PVA

2.4. Evaluation Indicators and Training Environment Setting

Deep learning has various evaluation metrics. In order to quantitatively analyze the performance of target detection algorithms, researchers have formulated many evaluation indicators, such as precision, recall, F1 score, frames per second (FPS), etc., [37]. Precision refers to the ratio of samples correctly classified by the classifier to the total number of samples. It measures the proportion of predicted positive samples that are actually positive, and it is calculated as shown in Equation (2). Recall represents the ratio of positive samples correctly classified by the classifier to the total number of true positive samples. It indicates the ability to correctly identify all positive cases, and it is calculated as shown in Equation (3). F1 score is a metric that represents the weighted average of accuracy and recall, reflecting the overall performance of a classifier. A higher F1 score indicates better classifier performance. The F1 score can be calculated by using Equation (4), as shown in the paper. AP, on the other hand, stands for average precision and is another evaluation metric used in object detection tasks. The calculation of AP is outlined in Equation (5), as presented in the paper.

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

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

$$F1 = \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{4}$$

$$AP = \int_0^1 Precision(Recall)d(Recall) \tag{5}$$

The training experimental environment is configured as shown in Table 3. In order to ensure the reliability of the ablation study and the analysis of comparative models, the input size of the target detectors in this study was uniformly set at 640×640. Due to changes in the model structure, unfreeze training was employed without utilizing pre-trained weights. The epoch and batch size were set to 610 and 16, respectively. The study utilized SGD as the optimizer, with a Score_Threshold value set to 0.5.

Table 3. Experimental configuration.

CPU	GPU	CUDA	Pycharm	Pytorch	Numpy	Torchvision
i9-10900KF	RTX-3090	11.0	2020.1×64	1.7.1	1.21.5	0.8.2

3. Results

3.1. Ablation Study

In order to verify the design rationality of CURI-YOLOv7, an ablation study was designed in this study, as shown in Table 4.

Table 4. Effect of different modules on CURI-YOLOv7.

Backbone	SPPF&L-E	Neck	GFLOPs	Paras (M)	Weights (MB)	FPS (on Computer)
			13.181	6.014	23.1	83.07
√			6.669	3.726	14.3	95.12
√	√		5.877	3.181	12.2	104.29
√	√	√	1.976	1.018	3.98	128.83

The backbone in the table refers to the MobileOne-block-based backbone constructed by CURI-YOLOv7 and discussed in Section 2.2.1. The abbreviations SPPF and L-E represent the two improvements discussed in Section 2.2.2. "Neck" in Table 4 represents the design of the neck network and the fusion of GSConv and depthwise separable convolution into the neck, as discussed in Sections 2.2.3 and 2.2.4. Since both are improvements pertaining to the neck, they are discussed together in the ablation study.

The experimental data reveal that the parameters and FPS of the YOLOv7tiny model are 6.014 M and 83.07, respectively, which are characterized by a large number of parameters and low computing speed. By gradually incorporating the improvement modules discussed earlier, Table 3 demonstrates that CURI-YOLOv7, with all improvements added or replaced, exhibits significant enhancements. Specifically, compared to YOLOv7tiny, CURI-YOLOv7 showcases an 85.01% reduction in GFLOPs, an 83.07% reduction in the number of parameters, an 82.77% reduction in model weight size, and a 55.09% increase in FPS on a computer in the experimental environment.

3.2. The Comparation of CURI-YOLOv7 on Embedded Device

In this study, we selected Faster-Rcnn [38], SSD [39], YOLOv5s, YOLOv7tiny, and YOLOv8n for conducting comparative experiments alongside the CURI-YOLOv7 proposed in this paper. As YOLOv8n necessitates pre-trained weights to ensure the accurate computation of average accuracy (mAP); it was trained using pre-trained weights. The training process included a freezing training phase with a freeze epoch of 50 and an unfreeze epoch of 610 for optimal results.

The operational flow chart testing on the embedded device is presented in Figure 9. The process begins with training five comparison models and CURI-YOLOv7 separately on the computer using the processed dataset. After training, the weights obtained from the computer training process are saved. The results of training the six networks on the computer are summarized in Table 5. After the completion of training, the weights file is imported into the embedded device Jetson Xavier Nx, which is deployed with the PyCharm and PyTorch environment. All images within the test set were employed, and the detection of an individual image was conducted 100 times consecutively for each FPS test on the embedded device. This meticulous approach was adopted to accurately ascertain the detection speed of the images.
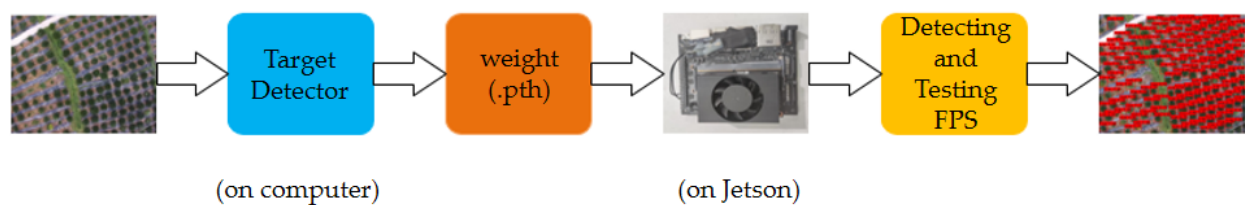

Figure 9. Flow chart for embedded device testing.

Table 5. Results of six models on a computer.

Models	GFLOPs	Paras (M)	Weights (MB)	F1	Recall (%)	Precision (%)	mAP@0.5 (%)
Faster-Rcnn	948.122	28.275	108	0.82	82.37	81.38	82.33
SSD	273.174	23.612	90.6	0.90	97.03	83.19	93.86
YOLOv5s	16.477	7.064	27.1	0.92	94.51	89.62	96.93
YOLOv7tiny	13.181	6.014	23.1	0.92	93.40	90.83	96.97
YOLOv8n	8.194	3.011	11.6	0.92	93.26	91.53	97.46
CURI-YOLOv7	1.976	1.018	3.98	0.86	83.70	89.02	90.34

The average FPS values obtained from above tests are shown in Figure 10. The results indicate that the two classical detectors, Faster R-CNN and SSD, exhibit generally slower average detection speeds. Faster R-CNN achieves an average detection speed of only 0.77, while SSD achieves 1.99 FPS. These speeds fall far below the average speed required for this study. With an average detection speed of 27.01 FPS on embedded devices, CURI-YOLOv7 delivers excellent performance and an 88.61% improvement over the original YOLOv7tiny.

3.3. The Comparison of CURI-YOLOv7 on Computer

Due to the poor performance of Faster R-CNN and SSD on embedded devices, as observed in Section 3.2, as well as their suboptimal mAP@0.5 results on the computer, it was determined that they were not suitable for comparison in this study. Additionally, their high GFLOPs and parameter counts do not meet the real-time image processing requirements of embedded devices. This is compounded by inadequate FPS. So, this section solely focuses on the performance comparison between the three YOLO series target detectors and the CURI-YOLOv7.

Figure 11 shows the PR curves of the four models. The horizontal coordinate is recall and the vertical coordinate is precision. The PR curves are generated by calculating a series of precision and recall points at different thresholds. The PR curve coverage areas for YOLOv8n, YOLOv5s, YOLOv7tiny, and CURI-YOLOv7 are 0.97, 0.96, 0.96, and 0.90, respectively. CURI-YOLOv7 does not cover as much area as the other three models, but also has a coverage area of 0.90. This suggests that while the average accuracy of CURI-YOLOv7 may be slightly lower than that of the comparison models, it is still substantial enough for the purposes of this study.
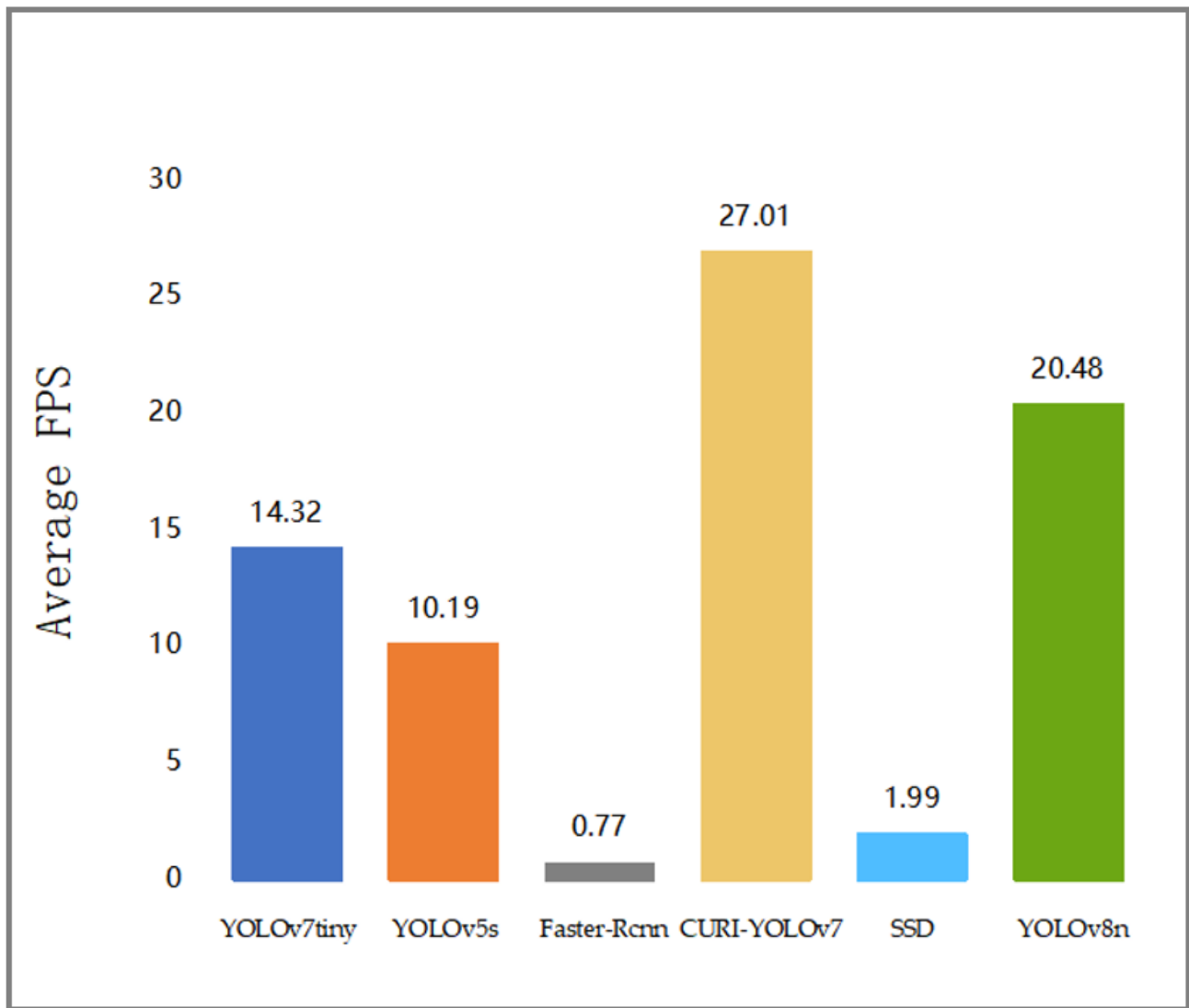
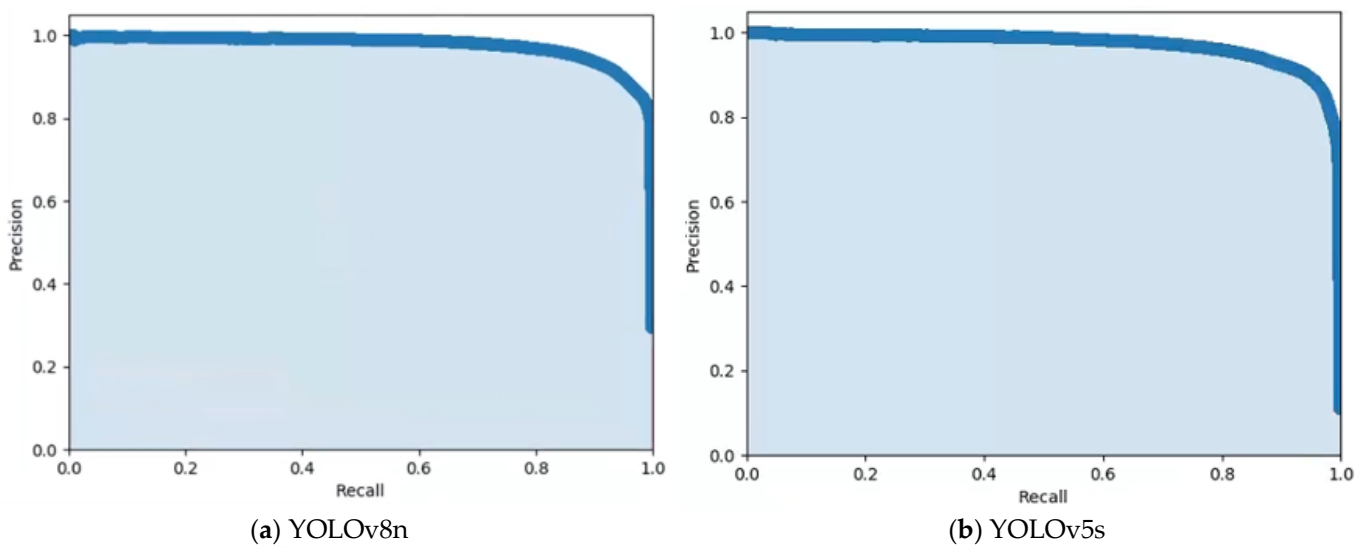**Figure 10.** FPS test results for embedded devices.

(**a**) YOLOv8n

(**b**) YOLOv5s

Figure 11. *Cont.*

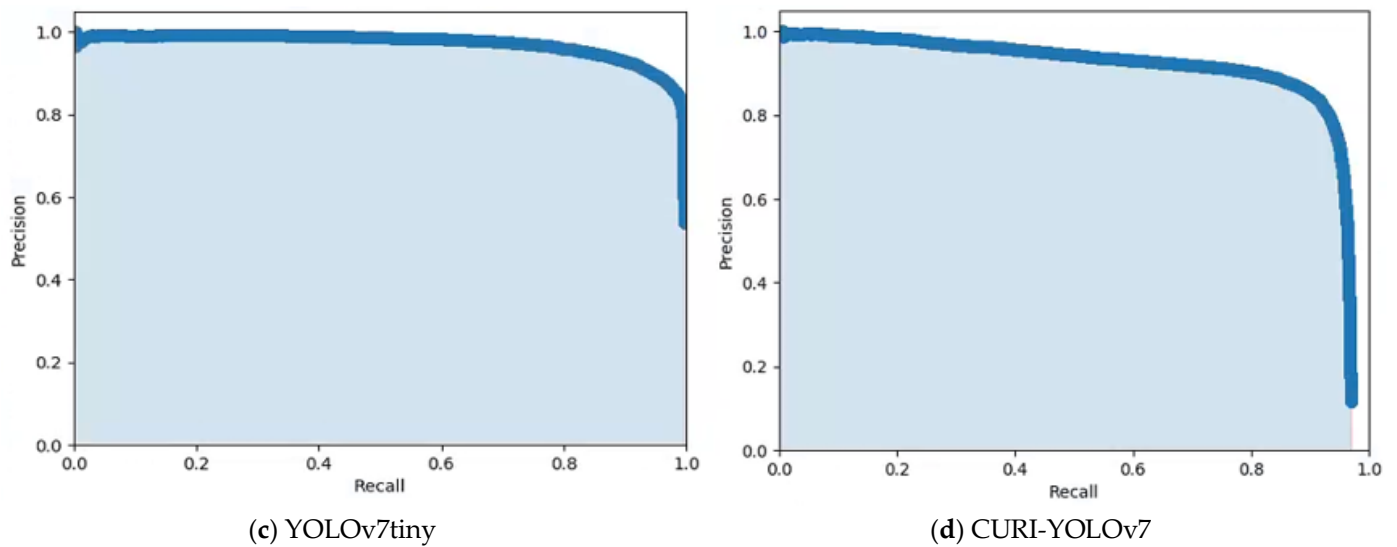**(c)** YOLOv7tiny　　　　　　　　　　　　　　　　**(d)** CURI-YOLOv7

Figure 11. Precision–recall (PR) curves for the four models.

Figure 12 illustrates the loss function curves of the four models. An examination of the loss function indicates that all four models converge within a localized region after approximately 400 iterations. This observation underscores that the design of CURI-YOLOv7 is comparably rational and viable, akin to other target detectors within the YOLO series. Moreover, it is notable that the loss function of CURI-YOLOv7 is the most minimized among the four models during the training process, converging within the range of 0 to 0.5.
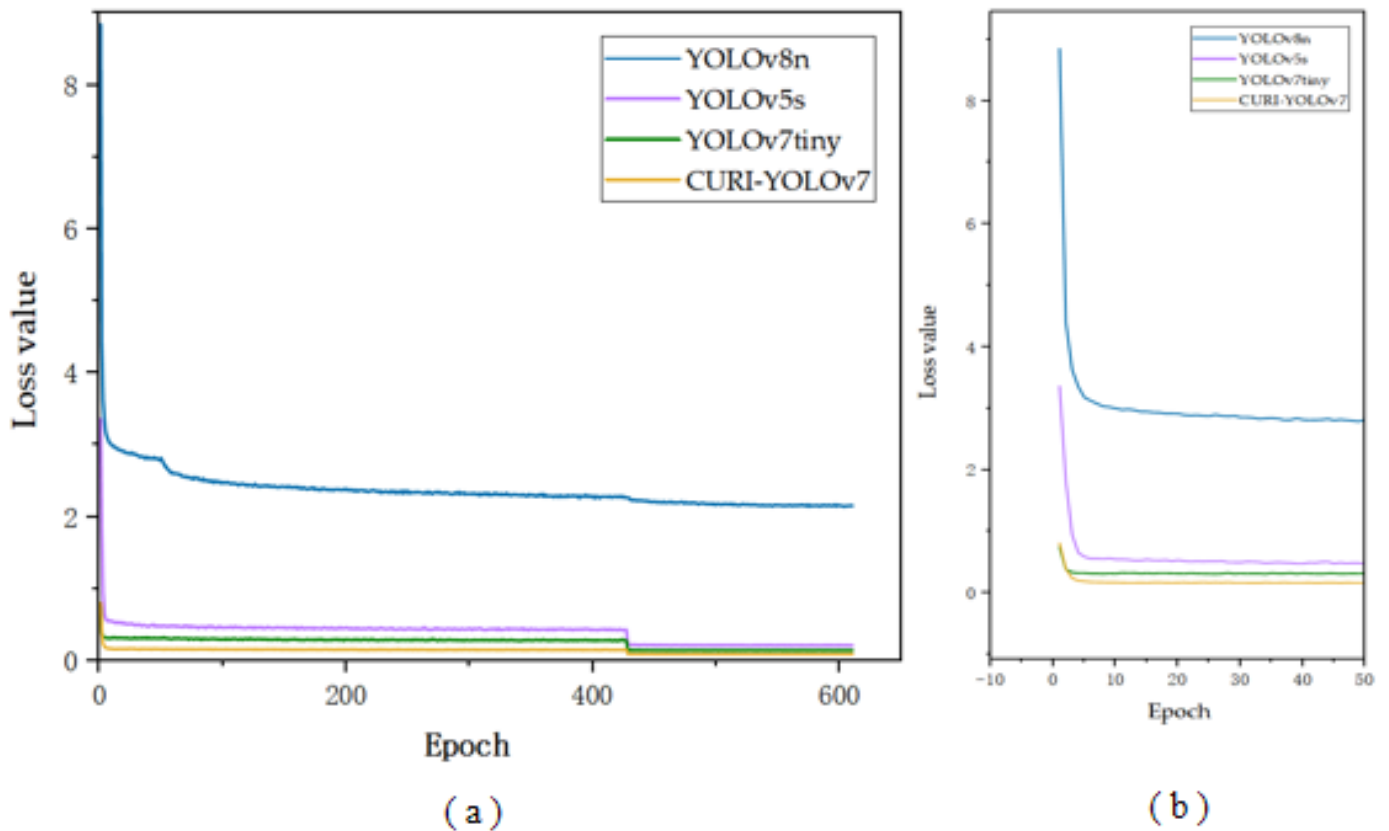

(a)　　　　　　　　　　　　　　　　　　　　　　　　**(b)**

Figure 12. Loss curves: (**a**) complete loss functions for the four models; (**b**) local zooms at 0 to 50 epochs.

In order to ensure algorithm robustness and adapt deep learning algorithm models to different environments [40], in this study, attention heat maps are employed to substantiate the robustness of CURI-YOLOv7. Figure 13 illustrates the attentional heat map outputs for the four models using three different test datasets. Coordinates and feature heat maps in essence output information on the location of key points [41] for the purpose of data visualization. In this study, the attentional heat map is generated by extracting the confidence of the predicted maximum value from the output prediction features and multiplying the two values to produce a visual representation. The output of YOLOv8n shows the best results among three scenarios, while CURI-YOLOv7 exhibits inferior results compared to YOLOv8n. However, CURI-YOLOv7 is still able to accurately focus attention on the correct position of the fruit tree.
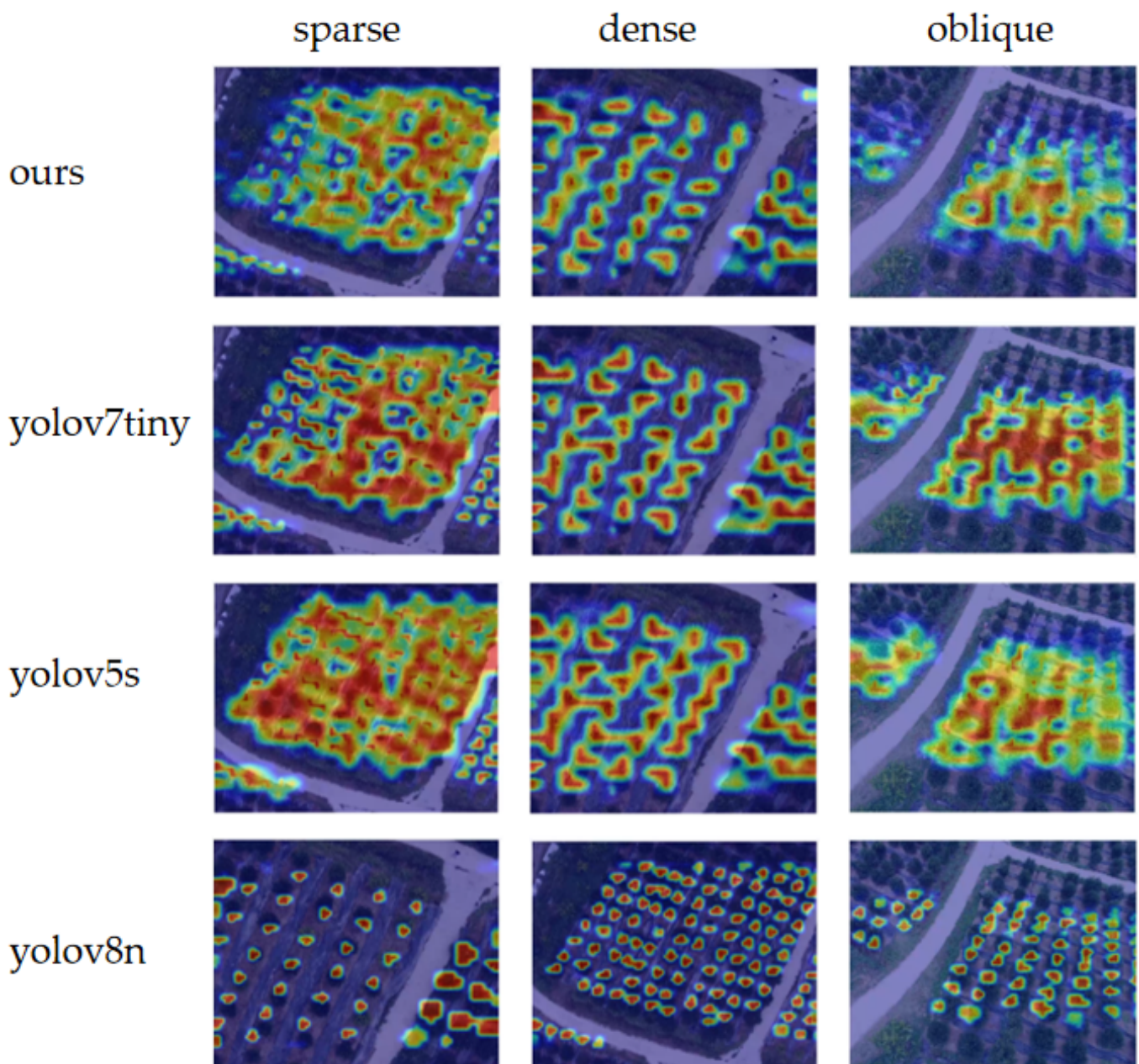

Figure 13. Heat map of attention for four target detectors in three environments.

4. Discussion

In this paper, a lightweight target detector named CURI-YOLOv7 is proposed for deployment on embedded devices. The algorithm is systematically compared and analyzed against five other models, including Faster-Rcnn, SSD, YOLOv5s, YOLOv7tiny, and YOLOv8n. This comparison serves to demonstrate the rationality and effectiveness of the design of CURI-YOLOv7. The experimental results affirm the advantages of CURI-YOLOv7. It achieves significant reductions in both GFLOPs and parameter quantities, resulting in a noteworthy improvement in frames per second (FPS). As a result, it is adept at carrying out real-time target detection when executed on embedded devices. This achievement effectively caters to the need for recognizing individual trees in UAV aerial images of citrus orchards while adhering to the limitations of embedded devices. Importantly, this solution effectively addresses the long-standing challenges related to unwieldy model sizes and slow image processing speeds. The domain of accurate orchard management has long been a focal point of research interest. The real-time remote sensing target monitoring facilitated by the integration of CURI-YOLOv7 into embedded devices emerges as a significant advancement. This innovation has the potential to extract relevant information from citrus tree imagery in real time, thereby underscoring the practical significance of the study.

This study does exhibit certain limitations. The dataset generated focuses exclusively upon UAV remote sensing of citrus trees, and consequently, the proposed model's applicability may not universally extend to other orchard remote sensing scenarios. Evaluating the outcomes, CURI-YOLOv7 attains a citrus tree target detection mean average precision (MAP) exceeding 90%. While marginally trailing behind other target detection algorithms within the YOLO series, this performance variation is noted.

In the realm of future endeavors, the inference speed of CURI-YOLOv7 will be subject to further reduction through techniques such as knowledge distillation and model pruning. These efforts are poised to establish a robust groundwork for subsequent investigations centered on the realm of automated UAV navigation.

5. Conclusions

This study proposes a lightweight YOLOv7tiny target detector CURI-YOLOv7 for the UAV aerial photography of citrus trees based on embedded devices. A dataset of citrus trees from UAV remote sensing imagery was constructed for training. The backbone is designed based on the MobileOne-block and depthwise separable convolution; the original spatial pyramid pooling is replaced by the SPPF, and the Lightweight-ELAN structure is constructed based on the ELAN structure. The neck is redesigned by removing the smaller target detection layer and fusion depth-separable convolution and GSConv to construct the proposed CURI-YOLOv7. The results show that the GFLOPs = 1.976, parameters = 1.018 M, weights = 3.98 MB, and mAP = 90.34% for CURI-YOLOv7. When executed on a computer, CURI-YOLOv7 displays a substantial enhancement in frames per second (FPS), at 128.83. Notably, on an embedded device, CURI-YOLOv7 excels with an impressive FPS improvement, achieving 27.01. This reinforces its adeptness for real-time processing on embedded devices. The progression of CURI-YOLOv7 establishes a fundamental basis for the subsequent advancement of low-altitude remote sensing technology in citrus orchard management and precision agricultural aviation.

Author Contributions: Conceptualization, Y.Z. and X.F.; methodology, Y.Z. and X.F.; validation, Y.Z. and X.F.; formal analysis, X.F. and J.G.; investigation, Y.Z., X.F. and J.G.; resources, X.F. and H.T.; data curation, X.F., J.G. and H.T.; writing—original draft preparation, Y.Z., X.F. and J.G.; writing—review and editing, Y.Z., L.W., H.T. and K.Y.; visualization, X.F. and L.W.; supervision Y.L.; project administration, Y.Z. and Y.L.; funding acquisition, Y.Z. and Y.L. All authors have read and agreed to the published version of the manuscript.

Data Availability Statement: The data presented in this study are available on request from the corresponding author.

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

References

1. Zhou, Z.; Ming, R.; Zang, Y.; He, X.; Luo, X.; Lan, Y. Development status and countermeasures of agricultural aviation in China. *Trans. Chin. Soc. Agric. Eng.* **2017**, *33*, 1–13.
2. Wang, L.; Lan, Y.; Zhang, Y.; Zhang, H.; Tahir, M.N.; Ou, S.; Liu, X.; Chen, P. Applications and prospects of agricultural unmanned aerial vehicle obstacle avoidance technology in China. *Sensors* **2019**, *19*, 642. [CrossRef] [PubMed]
3. Zhang, D.; Lan, Y.; Chen, L.; Wang, X.; Liang, D. Current status and future trends of agricultural aerial spraying technology in China. *Trans. Chin. Soc. Agric. Mach.* **2014**, *45*, 53–59.
4. Shi, Z.; Liang, Z.; Yang, Y.; Guo, Y. Status and prospect of agricultural remote sensing. *Trans. Chin. Soc. Agric. Mach.* **2015**, *46*, 247–260.
5. Nie, J.; Yang, B. Monitoring Method of Crop Growth on a Large Scale Basedon Remote Sensing Technology. *Comput. Simul.* **2020**, *37*, 386–389+394.
6. Shan, Y. Present situation development trend and countermeasures of citrus industry in China. *J. Chin. Inst. Food Sci. Technol.* **2008**, *8*, 1–8.
7. Dai, J.; Xue, J.; Zhao, Q.; Wang, Q.; Cheng, B.; Zhang, G.; Jiang, L. Extraction of cotton seedling growth information using UAV visible light remote sensing images. *Trans. Chin. Soc. Agric. Eng.* **2010**, *36*, 63–71.
8. Deng, J.; Ren, G.; Lan, Y.; Huang, H.; Zhang, Y. Low altitude unmanned aerial vehicle remote sensing image processing based on visible band. *J. South China Agric. Univ.* **2016**, *37*, 16–22.
9. Lan, Y.; Deng, X.; Zeng, G. Advances in diagnosis of crop diseases, pests and weeds by UAV remote sensing. *Smart Agric.* **2019**, *1*, 1–19.
10. Chen, Y.; Zhao, C.; Wang, X.; Ma, J.; Tian, Z. Prescription map generation intelligent system of precision agriculture based on knowledge model and WebGIS. *Sci. Agric. Sin.* **2007**, *pp*, 1190–1197.
11. Hao, L.; Shi, L.; Cao, L.; Gong, J.; Zhang, A. Research status and prospect of cotton terminal bud identification and location technology. *J. Chin. Agric. Mech.* **2018**, *39*, 72–78.
12. Tian, H.; Fang, X.; Lan, Y.; Ma, C.; Huang, H.; Lu, X.; Zhao, D.; Liu, H.; Zhang, Y. Extraction of Citrus Trees from UAV Remote Sensing Imagery Using YOLOv5s and Coordinate Transformation. *Remote Sens.* **2022**, *14*, 4208. [CrossRef]
13. Shu, M.; Li, S.; Wei, J.; Che, Y.; Li, B.; Ma, T. Extraction of citrus crown parameters using UAV platform. *Trans. Chin. Soc. Agric. Eng.* **2021**, *37*, 68–76.
14. Sun, Y.; Han, J.; Chen, Z.; Shi, M.; Fu, H.; Yang, M. Monitoring method for UAV image of greenhouse and plastic-mulched Landcover based on deep learning. *Trans. Chin. Soc. Agric. Mach.* **2018**, *49*, 133–140.
15. Wang, B.; Yang, G.; Yang, H.; Gu, J.; Zhao, D.; Xu, S.; Xu, B. UAV images for detecting maize tassel based on YOLO_X and transfer learning. *Trans. Chin. Soc. Agric. Eng.* **2022**, *38*, 53–62.
16. Bao, W.; Xie, W.; Hu, G.; Wang, X.; Su, B. Wheat ear counting method in UAV images based on TPH-YOLO. *Trans. Chin. Soc. Agric. Eng.* **2023**, *39*, 155–161.
17. Zhang, Y.; Lu, X.; Li, W.; Yan, K.; Mo, Z.; Lan, Y.; Wang, L. Detection of Power Poles in Orchards Based on Improved Yolov5s Model. *Agronomy* **2023**, *13*, 1705. [CrossRef]
18. Luo, X.; Wu, Y.; Zhao, L. YOLOD: A Target Detection Method for UAV Aerial Imagery. *Remote Sens.* **2022**, *14*, 3240. [CrossRef]
19. Zhu, Y.; Zhou, J.; Yang, Y.; Liu, L.; Liu, F.; Kong, W. Rapid Target Detection of Fruit Trees Using UAV Imaging and Improved Light YOLOv4 Algorithm. *Remote Sens.* **2022**, *14*, 4324. [CrossRef]
20. Basso, M.; Stocchero, D.; Ventura Bayan Henriques, R.; Vian, A.L.; Bredemeier, C.; Konzen, A.A.; Pignaton de Freitas, E. Proposal for an Embedded System Architecture Using a GNDVI Algorithm to Support UA V-Based Agrochemical Spraying. *Sensors* **2019**, *19*, 5397. [CrossRef]
21. Ki, M.; Cha, J.; Lyu, H. Detect and Avoid System Based on Multi Sensor Fusion for UAV. In Proceedings of the 2018 International Conference on Information and Communication Technology Convergence (ICTC), Jeju Island, Republic of Korea, 17–19 October 2018; pp. 1107–1109.
22. Chen, L.; Li, S.; Bai, Q.; Yang, J.; Jiang, S.; Miao, Y. Review of image classification algorithms based on convolutional neural networks. *Remote Sens.* **2021**, *13*, 4712. [CrossRef]
23. Liu, F.; Liu, Y.K.; Lin, S.; Guo, W.Z.; Xu, F.; Zhang, B. Fast recognition method for tomatoes under complex environments based on improved YOLO. *Trans. CSAM* **2020**, *51*, 229–237.
24. labelImg. Available online: https://github.com/tzutalin/labelImg (accessed on 15 June 2023).
25. Vasu PK, A.; Gabriel, J.; Zhu, J.; Tuzel, O.; Ranjan, A. An improved one millisecond mobile backbone. *arXiv* **2022**, arXiv:2206.04040.
26. yolov5. Available online: https://github.com/ultralytics/yolov5 (accessed on 7 July 2023).
27. Li, J.; Ye, J. Edge-YOLO: Lightweight Infrared Object Detection Method Deployed on Edge Devices. *Appl. Sci.* **2023**, *13*, 4402. [CrossRef]

28. Wang, C.Y.; Bochkovskiy, A.; Liao, H.Y.M. YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. *arXiv* **2022**, arXiv:2207.02696.

29. Long, Y.; Yang, Z.; He, M. Recognizing apple targets before thinning using improved YOLOv7. *Trans. Chin. Soc. Agric. Eng.* **2023**, *39*, 191–199.

30. Li, Z.; Yang, F.; Hao, Y. Small target detection algorithm for aerial photography based on residual network optimization. *Foreign Electron. Meas. Technol.* **2022**, *41*, 27–33.

31. Li, H.; Li, J.; Wei, H.; Liu, Z.; Zhan, Z.; Ren, Q. Slim-neck by GSConv: A better design paradigm of detector architectures for autonomous vehicles. *arXiv* **2022**, arXiv:2206.02424.

32. Howard, A.G.; Zhu, M.; Chen, B.; Kalenichenko, D.; Wang, W.; Weyand, T.; Andreetto, M.; Adam, H. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv* **2017**, arXiv:1704.04861.

33. Zhang, M.; Gao, F.; Yang, W.; Zhang, H. Wildlife Object Detection Method Applying Segmentation Gradient Flow and Fea ture Dimensionality Reduction. *Electronics* **2023**, *12*, 377. [CrossRef]

34. Wu, T.; Zhang, Z.; Liu, Y.; Pei, W.; Chen, H. A lightweight small object detection algorithm based on improved SSD. *Infrared. Laser Eng.* **2018**, *47*, 703005.

35. Kong, W.; Li, W.; Wang, Q.; Cao, P.; Song, Q. Design and implementation of lightweight network based on improved YOLOv4 algorithm. *Comput. Eng.* **2022**, *48*, 181–188.

36. Caba, J.; Díaz, M.; Barba, J.; Guerra, R.; de la Torre, J.A.; López, S. Fpga-based on-board hyperspec tral imaging compression: Benchmarking performance and energy efficiency against gpu implementations. *Remote Sens.* **2020**, *12*, 3741. [CrossRef]

37. Wang, C.; Wang, Q.; Wu, H.; Zhao, C.; Teng, G.; Li, J. Low-Altitude Remote Sensing Opium Poppy Image Detection Basedon Modified YOLOv3. *Remote Sens.* **2021**, *13*, 2130. [CrossRef]

38. Ren, S.; He, K.; Girshick, R.B.; Sun, J. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *Ieee Trans. Pattern Anal. Mach. Intell.* **2015**, *39*, 1137–1149. [CrossRef]

39. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.-Y.; Berg, A.C. SSD: Single Shot MultiBox Detector. *arXiv* **2015**, arXiv:1512.02325.

40. Zhang, Y.; Xiao, W.; Lu, X.; Liu, A.; Qi, Y.; Liu, H.; Shi, Z.; Lan, Y. Method for detecting rice flowering spikelets using visible light images. *Trans. Chin. Soc. Agric. Eng.* **2021**, *37*, 253–262.

41. Liu, Y.; Li, J.; Zhang, J.; Xu, Z. Research progress of two-dimensional human pose estimation based on deep learning. *Comput. Eng.* **2021**, *47*, 1–16.