agronomy

MDPI

Article

An Improved Rotating Box Detection Model for Litchi Detection in Natural Dense Orchards

Bin Li [1], Huazhong Lu [1], Xinyu Wei [1], Shixuan Guan [2], Zhenyu Zhang [2], Xingxing Zhou [1] and Yizhi Luo [1,*]

[1] Institute of Facility Agriculture, Guangdong Academy of Agricultural Sciences, Guangzhou 510640, China; ganli180@sina.com (B.L.); wxyscau@163.com (X.W.); zhouxingxing@gdaas.com (X.Z.)
[2] College of Engineering, South China Agricultural University, Guangzhou 510642, China; 20233173013@stu.scau.edu.cn (S.G.); zzy1290037444@163.com (Z.Z.)
* Correspondence: luoyizhi@gdaas.cn; Tel.: +86-020-38815523

Abstract: Accurate litchi identification is of great significance for orchard yield estimations. Litchi in natural scenes have large differences in scale and are occluded by leaves, reducing the accuracy of litchi detection models. Adopting traditional horizontal bounding boxes will introduce a large amount of background and overlap with adjacent frames, resulting in a reduced litchi detection accuracy. Therefore, this study innovatively introduces the use of the rotation detection box model to explore its capabilities in scenarios with occlusion and small targets. First, a dataset on litchi rotation detection in natural scenes is constructed. Secondly, three improvement modules based on YOLOv8n are proposed: a transformer module is introduced after the C2f module of the eighth layer of the backbone network, an ECA attention module is added to the neck network to improve the feature extraction of the backbone network, and a 160 × 160 scale detection head is introduced to enhance small target detection. The test results show that, compared to the traditional YOLOv8n model, the proposed model improves the precision rate, the recall rate, and the mAP by 11.7%, 5.4%, and 7.3%, respectively. In addition, four state-of-the-art mainstream detection backbone networks, namely, MobileNetv3-small, MobileNetv3-large, ShuffleNetv2, and GhostNet, are studied for comparison with the performance of the proposed model. The model proposed in this article exhibits a better performance on the litchi dataset, with the precision, recall, and mAP reaching 84.6%, 68.6%, and 79.4%, respectively. This research can provide a reference for litchi yield estimations in complex orchard environments.

Keywords: litchi detection; oriented bounding box; transformer module; eca attention mechanism; small target detection

Citation: Li, B.; Lu, H.; Wei, X.; Guan, S.; Zhang, Z.; Zhou, X.; Luo, Y. An Improved Rotating Box Detection Model for Litchi Detection in Natural Dense Orchards. *Agronomy* **2024**, *14*, 95. https://doi.org/10.3390/agronomy14010095

Academic Editors: Ahmed Rady and Ahmed Kayad

Received: 29 November 2023
Revised: 21 December 2023
Accepted: 27 December 2023
Published: 30 December 2023

1. Introduction

Litchis (Litchi chinensis Sonn.) are a very popular tropical fruit that are widely grown around the world, including Central and South America, parts of Africa, and regions throughout Asia. Among them, China is the world's largest producer of litchi [1,2]. In 2022, the litchi planting area in mainland China was 7.8915 million acres, and the Litchi output was approximately 2.2227 million tons [3]. Litchi fruit degradation impacts their flavor as a result of the synthesis of lycheic acid triggered by prolonged exposure to elevated temperatures [4]. Therefore, predicting orchard litchi yields in advance is a critical issue. These predictions help farmers effectively plan their storage and harvesting schedules and enable them to develop early supply strategies.

In actual production, orchard managers usually use random sampling and manual counting to estimate orchard yields [5]. However, scattered litchi clusters, obstructions due to leaves and branches, and the complex lighting conditions of the natural environment lead to low-accuracy litchi identification [6,7]. Consequently, achieving real-time and accurate litchi identification in natural dense scenes is one of the core issues in improving the accuracy of litchi yield estimations.

In recent years, researchers have conducted significant amounts of research on litchi detection [8–10]. An improved linear discriminant analysis classifier was proposed for distinguishing green litchis in similar color backgrounds. This model introduced the idea of the "maximum margin" in SVM and optimized the threshold of the LDA classifier. The precision and recall rates of green litchi were 80.4% and 76.4%, respectively [11]. To solve the litchi occlusion problem, a litchi detection method based on monocular machine vision was introduced. This method separates single litchis under different occlusion conditions based on a combination of contrast limited adaptive histogram equalization (CLAHE), red/blue color mapping, the Otsu threshold, and morphological operations [12]. In [13], a depth-camera-based method for litchi fruit recognition in the natural environment was introduced. The above model design artificially introduces descriptors of litchi characteristics to improve the recognition accuracy of litchi fruits under complex occlusion conditions. However, dense litchi clusters are still challenging due to their scattered nature, leaf occlusion, and complex lighting conditions. However, dense litchi clusters are still a big challenge due to their scattered nature, leaf occlusion, and the complex lighting conditions.

With the rapid development of deep learning technology, excellent target detection algorithms such as YOLO, SSD, and Faster R-CNN have been applied to fields such as fruit and vegetable detection [14–19]. For example, the k-means++ clustering algorithm has been used to optimize bounding box settings, reduce the number of network layers, and improve litchi detection in dense scenes [20]. In [21], a convolutional block attention module was added to each C3 module of the network backbone to enhance the network's ability to extract important feature information. Wang et al. [22] used ShuffleNetv2 as the backbone network for litchi fruit detection. In the feature fusion stage, they introduced the CBAM module to further refine the effective litchi feature information. The above work has achieved good results in the identification of litchi fruits in dense scenes. However, the following issues still exist:

1 Background interference: Since litchi often grow at an angle or in clusters, if horizontal bounding boxes are used to select litchi, a large amount of background will appear in the frame and adjacent frames will overlap, affecting the litchi detection accuracy.
2 Litchis growing in a natural environment exist in diverse and intricate backgrounds and are often obscured by branches or leaves.
3 The multi-scale recognition accuracy is low. With a large viewing angle, the size of litchi fruits varies greatly, and the litchi detection accuracy is low.

2. Related Work

At present, litchi detection methods can be roughly divided into two categories. One is the traditional object detection method based on artificially designed features, including color, shape, and contour characteristics; the other is a target detection method based on a combination of deep learning [23].

Regarding traditional target detection methods, Guo et al. [12] introduced a combination of color chromatic mapping, morphology operation, and other strategies to extract individual litchi from the litchi foreground region. Xiong et al. [2] used an improved fuzzy clustering algorithm (FCM) to first remove the background of litchi images at night and then used the Otsu algorithm to segment the fruits and main stems of litchi bunches. Finally, the picking point was obtained using the Harris corner point. The experimental results using this algorithm showed that the night recognition accuracy of litchi bunches was 93.75%, and the average recognition time was 0.516 s. Moreover, a matching algorithm to locate litchi based on a label template was discussed. In this method, litchi with a similar label template was matched according to the preset threshold by traversing a litchi label template of a left image in a right image to find optimal matching. The experimental results showed that the proposed recognition method could be robust against the influences of varying illumination and precisely recognizing litchi. The highest average recognition rate for unoccluded and partially occluded litchi was 98.8% and 97.5%, respectively [24]. Yu et al. proposed that a red-green-blue depth (RGB-D) camera can be used to estimate

fruit yield. In this method, depth images were introduced to segment redundant image information that excludes the effective pickup range of the robotic arm. Secondly, color and texture features were used to identify litchi fruits. The recognition accuracy of this method for green litchi and red litchi was 89.92% and 94.50%, respectively [13]. Compared with apples and pears, the detection results were easily affected by light changes due to the uneven surface of the fruit. In addition, litchi detection methods based on traditional target detection, such as color and texture features, have low robustness in dense scenes and occlusions.

Regarding litchi detection research based on deep learning, Peng et al. [25] proposed a novel network model that used the feature pyramid to retain the shallow features of litchis and cut down the model size. Wu et al. [7] proposed a litchi detection algorithm based on the YOLOv4 network, using the K-means++ algorithm to select the appropriate size for litchis, and changed the feature map to account for the small and dense litchi targets, but the improved algorithm still missed detection in large scenes. Furthermore, a long-close distance coordination control strategy was proposed based on an RGB-depth camera combined with a point cloud map. Compared with existing studies, this strategy took into account the advantages and disadvantages of depth cameras. By experimenting with the complete process, the density-clustering approach in long distances was used to classify different bunches at a closer distance, while a success rate of 88.46% was achieved from fruit-bearing branch locating [26]. Moreover, the attention mechanism was introduced to improve the model's ability to extract features. An improved fully convolutional single-stage object detection (FCOS) network was used for litchi leaf disease detection. In this method, the central moment pooling attention (CMPA) mechanism is introduced to enhance the characteristics of litchi leaf diseases and insect pests. The test results showed that the average detection accuracy of images of five kinds of litchi leaf diseases and insect pests was 91.3%, and the model parameter size was 17.65 M [27]. The method based on HBBs can realize the accuracy of litchi in complex lighting scenes. However, for occlusion between litchi between litchi research, the annotation based on HBBs in practical application is very limited.

Rotating frame detection is a novel object detection method. This method adjusts the angle of the target frame so that the frame can better cover the target area, effectively avoid interference from background features, and improve the network's ability to learn target area features. In this way, when the litchi is blocked or squeezed, the detection method based on the rotating frame can reflect the shape of the litchi.

In order to achieve the accurate detection and yield estimation of litchis in natural scenes, in this work, an innovative litchi detection model based on YOLOv8 and rotating boxes is proposed, reducing the background image and adjacent boxes due to frame selection. Regarding the impact of overlap on feature extraction, a transformer module is introduced to avoid losing feature information in the network and solve the background interference problem, and it is combined with the Efficient Channel Attention Network (ECA) attention mechanism and then added to the algorithm model. Thus, the algorithm model can more accurately identify litchis. In addition, an additional detection head is introduced into the algorithm model, which has excellent small-target detection capabilities for litchis.

3. Materials and Methods

3.1. Experimental Area and Data Acquisition

As in shown in Figure 1, all data in the experiment were collected at a litchi orchard in Maoming City, Guangdong Province, China (23°15′08″ N, 113°37′12″ E), and at the Litchi Garden of the Fruit Tree Research Institute at the Guangdong Academy of Agricultural Sciences (21°78′15″ N, 111°16′51″ E). The studied litchi variety was Guiwei. In this study, the imaging equipment was an industrial camera. Images were collected from 9:00 to 5:00 p.m. from mid-May to early June 2022. The image resolution was 4032 × 3024. In particular, 43 litchi trees were selected; the height of the litchi trees ranged from 3 to

5 m. In order to improve the model's generalization ability, the distance between the camera and the litchi tree was 0.5–5 m, and blurred images generated during motion were removed. The final dataset contained 1228 images, and the dataset was divided into a training set and a validation set in a ratio of 8:2. The image annotation methods and results are shown in Figure 2. It can be observed that the rotated rectangular target box contains fewer redundant details, and angle information is added to represent the orientation of the original horizontal bounding box.

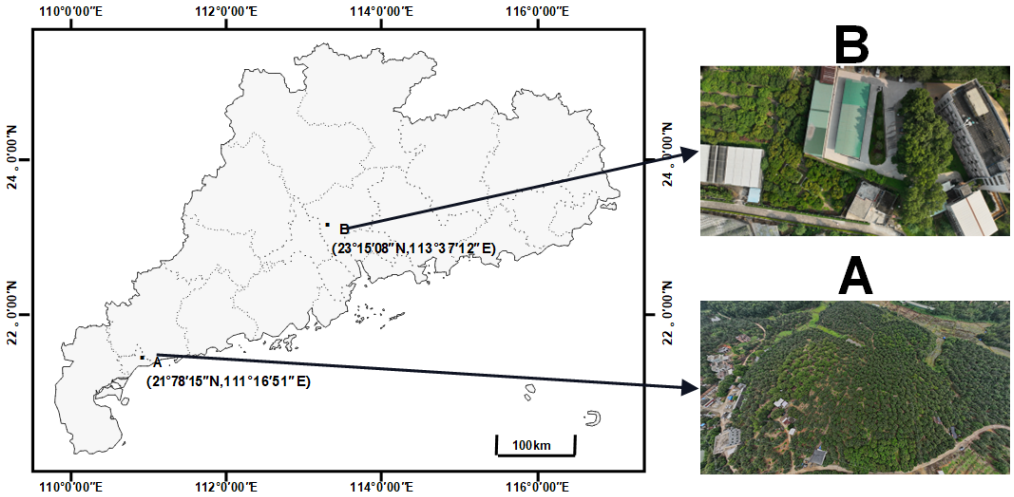

Figure 1. Experiment, (**A**) Aerial photo of collection location A, (**B**) Aerial photo of collection location B area.

Figure 2. Data preprocessing and annotation.

3.2. Data Annotation

Litchi detection typically is achieved by determining objects of interest with HBB annotation. However, the shape and size of litchi can influence their appearance in real-world environments. If HBBs are utilized for the task of litchi detection, this approach introduces a significant amount of background into the frame and may result in a high number of overlapping adjacent frames [28]. The accuracy of the model is diminished. Consequently, compared with HBB annotation, the OBB is suitable for litchi detection. In particular, the data processing link is to use the interactive labeling tool rolabelimg (https://github.com/cgvict/roLabelImg, accessed on 28 November 2023) to label the litchi image, refer to the DOTA dataset type, and export the .json file.

3.3. Dense Litchi Detection Network Architecture

In view of the characteristics of litchis in the images, such as their small size, large resolution differences, and insufficient feature information, this paper proposes an improved rotating box detection model for detecting litchi in natural dense orchards. In this model, the detection head of YOLOv8 is replaced with rotating box object detection [29]. At the same time, a transformer module and an ECA attention mechanism module are used in the

backbone neural network [30–32]. As shown in Figure 3, the model proposed in this article consists of three parts: a backbone, a neck, and a head.

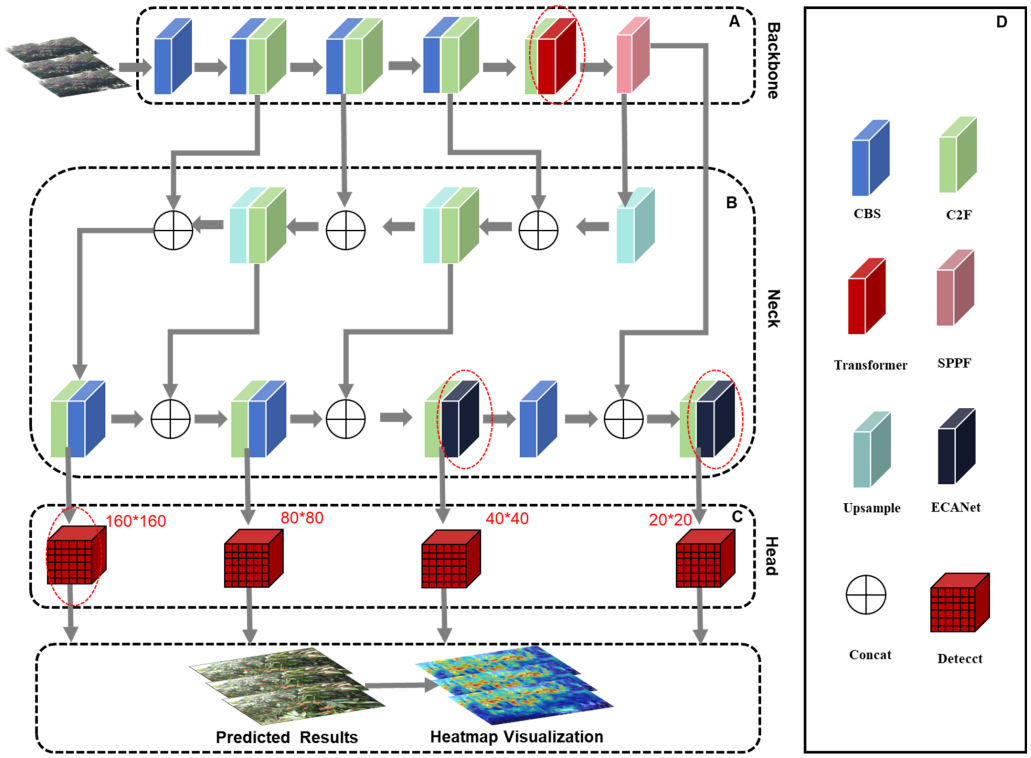

Figure 3. Overall framework of litchi recognition. (**A**) The backbone of the model, (**B**) the neck of the model, (**C**) the head of the model, and (**D**) the name corresponding to each module. Red oval dotted boxes indicate areas for improvement.

3.3.1. Network Structure of YOLOv8

The traditional YOLOv8 network consists of a backbone network (backbone) for feature extraction, a neck network (neck) for feature fusion, and a detection head (head) for predicting. The backbone has a cross-stage local network structure to reduce the calculation amount and enhance the gradient. A spatial pyramid pooling module is used to better extract spatial features. In the neck, the convolution structure in the PAN-FPN upsampling stage in YOLOv5 has been removed, and the downsampling operation is performed first, followed by the upsampling operation. Replacing the C3 module with the C2f module decreases the model weight and makes it more adaptable to different sized and shaped targets [33–35]. In the head part, the current mainstream decoupled head structure (decoupled head) is adopted, which effectively reduces the number of parameters and computational complexity and enhances the model's generalization ability and robustness. Finally, the bounding box loss is calculated by the Conv2d block. Different from previous versions of YOLO, YOLO-V8 uses Anchor-free instead of Anchor-base to directly predict the position and size of the anchor box, thereby further improving detection speed and accuracy of the model. The advantages of the YOLOv8 model can be summarized as follows:

- Backbone: The backbone is mainly composed of a CBS module, a C2F module, and an SPPF module. The CBS module consists of a convolution, batch normalization, and normalization SiLU functions. The main function of BN is to maintain the same distribution in each ANN layer to avoid gradient disappearance in network training; the CBS module compresses and expands the feature information by changing the number of feature channels, thereby improving and balancing the calculation speed and accuracy of the ANN; the C2f module is a network component used to extract deep feature

information. The latter can be embedded at any position or replace any convolutional layer to enhance the backbone's performance.

- Neck: The convolution structure in the PAN-FPN upsampling stage in YOLOv5 is removed, and the downsampling operation is performed first, followed by the upsampling operation. Replacing the C3 module with the C2f module reduced the weight and makes the model more adaptable to targets of different sizes and shapes.
- Head: The current mainstream decoupled-head structure is adopted to effectively reduce the parameter number and computational complexity and enhance the model's generalization ability and robustness. The design of using anchor-base to predict the position and size of the anchor box, as used previously in the YOLO series, has been abandoned, and instead an anchor-free detection method is used to directly predict the target's center point, width, and height. Reducing the number of anchor boxes further improves the model's detection speed and accuracy.
- Loss function: The model uses CIOU Loss as the error loss function to further improve the regression accuracy of the bounding box by minimizing the DFL. At the same time, the model adopts the Task Aligned Assigner sample allocation strategy, using the high-order combination of the classification score and the IOU as an indicator to guide the selection of positive and negative samples. This successfully aligns the high classification score and high IOU, effectively improving the model's detection accuracy.

3.3.2. The Transformer Module

A transformer module is an architecture of the attention mechanism, which has achieved great success in NLP, for example, in the currently famous artificial intelligence ChatGPT. A transformer module can improve the reasonable distribution of attention weights, so that the output of the attention layer contains representative information of different subspace encodings, and can also enhance a model's representation ability. Secondly, transformers' multi-scale feature fusion capability helps comprehensively utilize information at different scales and improves the performance and stability of target detection. Through position encoding and spatial relationship modeling, a model can more accurately locate targets and capture relative distance and angle information between targets [36,37].

A transformer module was introduced into the backbone of the YOLOv8 network to enhance the target detection accuracy. In particular, the transformer module in this study uses a multi-head self-attention mechanism instead of a single-head self-attention mechanism in order to enhance the focus on the connection between global features and its own position [38].

As shown in Figure 4, first, the litchi image feature information is input into the BN layer to standardize the distribution of feature information to improve the training speed, and it is then connected to a MHA module. This module is designed to enhance the algorithm's extraction of deep feature information from images and improve the model's accuracy when detecting images with multiple scales, small targets, and high noise levels. In order to suppress network overfitting, a dropout layer is added after the MHA module. At the same time, the input and feature information extracted through the MHA module is weighted and combined to make the feature information clearer. A batch normalization layer and a MLP are then connected to improve the neural network's nonlinear fitting and expression abilities and thus further improve its performance. Finally, all feature information is weighted and output to ensure the integrity of the feature information, thereby improving the detection accuracy of the neural network [24].

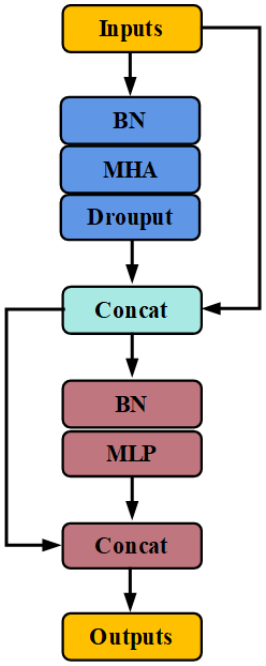

Figure 4. transformer.

3.3.3. Increased ECA-Net Mechanisms

Attention mechanisms have been widely used in the field of computer vision, especially in target detection tasks. There are currently two main types of attention mechanisms: channel attention mechanisms and spatial attention mechanisms. SElayer explicitly models cross-dimensional interactions to extract channel attention [39]. CBAM utilizes the semantic interdependence between spatial dimensions and channel dimensions in feature maps to establish cross-channel and cross-space information [40]. SGE groups the channel dimensions into multiple sub-features, improving the spatial division of different semantic sub-feature representations [41]. In the process of litchi recognition, in addition to branches or leaves blocking the fruit, there is also a large amount of complex background information, which is also one of the reasons for poor recognition accuracies [42].

In response to the above problems, in this study, the ECANet module is introduced into the YOLOv8 model by adding it to the neck of the module in the algorithm model, enhancing the model's feature extraction capability and thus improving its recognition accuracy. As is shown in Figure 5, the ECA module is used in front of 40 × 40 and 20 × 20 detection heads. In particular, in the experiment, the attention mechanism was not used before all detection heads due to the accuracy was not improved.

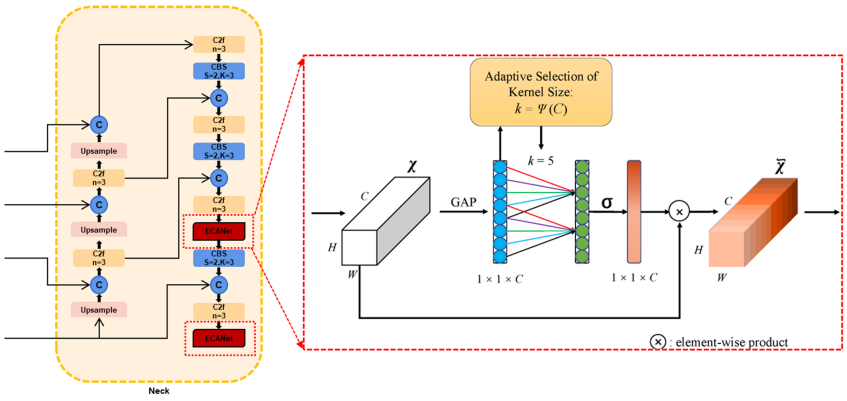

Figure 5. ECA model architecture.

In the ECA module, a local cross-channel interaction strategy is proposed without dimensionality reduction, which considers each channel and its k nearest neighbors after channel global average pooling (GAP) to capture local cross-channel interaction information. The ECA module first performs spatial feature compression on the input feature map to obtain a $1 \times 1 \times C$ feature map; secondly, it performs channel feature learning and captures cross-channel interactive information through one-dimensional convolution of the dynamic convolution kernel, which is adapted from the convolution kernel. The function formula is adaptively selected to determine the coverage of local cross-channel interactions; finally, the output feature channel weight vector generated by activation function calculation is multiplied channel by channel with the original input feature map to output a feature map with channel attention, thereby improving effective feature extraction. The ECA module avoids channel dimensionality reductions, allowing the model to more effectively learn channel attention. The module has a small number of parameters, which are only determined by its convolution kernel size k. Its formula is:

$$k = \varphi(n) \tag{1}$$

$$\varphi(n) = |\frac{\log_2(C)}{\gamma} + \frac{b}{\gamma}|_{odd} \tag{2}$$

where k represents convolution kernel size of 1D convolution, n represents the classes ($n = 1$), C represents channel dimension, and $|t|_{odd}$ indicates the nearest odd number of t. In this paper, we set γ and b to 2 and 1 throughout all the experiments, respectively.

3.3.4. Head Prediction Branch Improvements

Occlusion and large distances of litchi in images often result in large differences in litchi scales. Large targets at the front of images occupy most of the area, while individual small targets, especially the head area, account for less than 1% of the image size, and feature information is easily lost [43]. In [40], an additional transformer module detection head was added to a detection model, achieving good results for high-speed and low-altitude flights on datasets with densely packed objects with drastic scale changes. This provided inspiration for the improvements proposed in this article.

The detection layer of the original YOLOv8 outputs three feature maps of different sizes, namely 20×20, 20×20, and 80×80. When the litchi target is too small, it will not be detected. Here, we added a 1600×160 small-target-detection head to the original YOLOv8n model to enhance the sensitivity to smaller targets. We propose a model that initially extracts features from the second layer of the backbone network, uses Concat splicing to fuse the shallow feature context information extracted from the neck network structure, and finally uses the fourth detection head output at Layer 18 as a small target detection head. This improvement slightly increases the model's calculation amount, but it greatly improves the small-target-detection ability by obtaining more feature information from small targets. False and missed detections of objects at different scales are effectively reduced. The improved detection layer is shown in Figure 6. This study introduces a prediction head for detecting small objects and combines it with the three other prediction heads. This four-head structure helps mitigate the negative impact caused by changes in object scale. As shown in Figure 6, our new prediction head extracts information from low-level, high-resolution feature maps and is therefore more sensitive to small objects. Although adding an extra detection heads results in increased computational and memory costs, small-object detection is significantly improved.

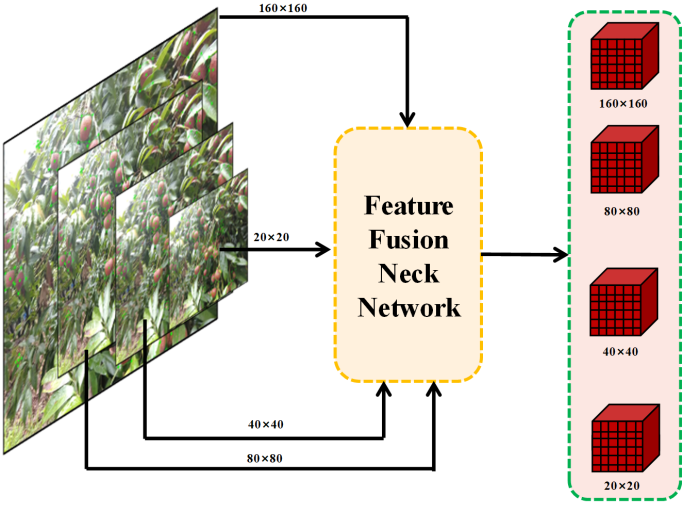

Figure 6. Improved detection layer.

4. Results and Discussion

4.1. Training Environment and Equipment Description

The algorithm was implemented on an Ubuntu 18.04 version system. The detailed hardware parameters of the device are shown in Table 1. During the data enhancement process of random affine transformations in the network, rotation and shear transformations that modify the object's angle were disabled before entering the network. The model parameters were initialized using weights pretrained on the COCO dataset and then fine-tuned. The model's hyperparameters were configured with a batch size of 16, a momentum factor of 0.95, and an initial learning rate of 0.01. The training process consisted of 300 iterations.

Table 1. Training environment and equipment description.

Configuration	Parameter
Image Resolution	4032 pixels × 3024 pixels (W × H)
Training framework	Python programming language, Pytorch framework
Pretrained model	ImageNet model
Operating system	Ubuntu18.04 version
Accelerated environment	CUDA11 and CUDNN 7
Development environment	Vscode
Computer configuration used in training and testing	Intel I7-87700K Processor, Huawei, China 32 GB RDIMM, 512 G Solid State Drive, 2 TB Mechanical Hard Drive, Graphics Card RTX3080Ti

4.2. Evaluation Metrics

The precision and recall rates and the mAP were adopted to evaluate the effectiveness of the litchi detection model in this paper [44–46]. The formulas are as follows:

$$Precision = \frac{TP}{TP + FP} \tag{3}$$

$$Recall = \frac{TP}{TP + FN} \tag{4}$$

$$AP = \int_0^1 Precision \times Recalldr \tag{5}$$

$$mAP = \frac{AP}{n} \tag{6}$$

where n represents the classes, $n = 1$.

4.3. Ablation Test

In order to evaluate the effectiveness of each improvement to the model proposed in this paper and their interactions, ablation experiments were conducted to analyze the causal relationship of each improved component. The experiment was divided into eight groups, namely, M0–M7, for training: M0 is the YOLOv8n model; M1 is M0 with an added transformer module behind the C2f module of the eighth layer in the backbone network of the YOLOv8n model; M2 is M0 with the ECA attention mechanism embedded into the neck network of the YOLOv8n model; M3 improves the head of the YOLOv8n model; M4–M6 are any two combinations of M0–M2, and M7 is the combination of all of the above three improved parts. In this study, the precision, recall, and mAP were used as indicators under the same experimental environment. The test results are shown in Table 2.

Table 2. Ablation test.

Classes	C2fTRS	ECA	Change Head	Parameters	Precision	Recall	mAP
M0	-	-	-	3380646M	75.7%	65.1%	74.0%
M1	✓	-	-	3316390M	81.4%	63.6%	74.7%
M2	-	✓	-	3380652M	80.0%	64.2%	74.6%
M3	-	-	✓	3802728M	81.7%	67.4%	77.4%
M4	✓	✓	-	-	82.9%	64.4%	76.1%
M5	✓	-	✓	-	85.0%	66.0%	78.2%
M6	-	✓	✓	-	84.1%	67.6%	78.5%
M7	✓	✓	✓	-	84.6%	68.6%	79.4%

M0 = YOLOv8n. M1 = YOLOv8n + C2fTRS. M2 = YOLOv8n + ECA. M3 = YOLOv8n + changed head. M4 = YOLOv8n + C2fTRS + ECA. M5 = YOLOv8n + C2fTRS + changed head. M6 = YOLOv8n + ECA + changed head. M7 = YOLOv8n + C2fTRS + ECA + changed head. Where ✓ represents the use of this module, - Indicates that this module is not used, the numbers with frames are optimal.

It can be seen from the ablation experiment results that, with the application of each improved module, the overall performance of the model improves. When the transformer module is added to the backbone network, the precision rate and mAP of the model increase by 7.5% and 0.9%, respectively, and the recall rate decreases slightly, indicating that the transformer module improves the precision of litchi detection; at the same time, the probability of missed litchi detections also increases. When the ECA attention mechanism is introduced into the cross-layer connection part of the detection based on M1, the precision rate and mAP of the model are further improved. In addition, it is obvious that the improved detection head has the highest contribution among the three improved modules, which inevitably slightly increases the amount of parameters and model calculations, but greatly improves the precision rate, the recall rate, and mAP.

4.4. Performance Comparison of Different Models

Litchi detection in naturally dense scenes has always been a challenge. Insufficient feature information, small litchi sizes, and occlusion can easily lead to missed and false detections. Four different advanced backbone network models were chosen for comparison in this paper, namely, MobilenetV3-Small, MobilenetV3-Large [39], ShuffleNetV2 [22], and GhostNet [47]; the comparison results are shown in Table 3.

Table 3. Performance comparison of different models.

Detector	Backbone	Precision	Recall	mAP
YOLOv8	mobilenetv3-small	76.9%	53.8%	67.5%
YOLOv8	mobilenetv3-large	80.2%	63.2%	73.8%
YOLOv8	shufflenetv2	78.8%	60.4%	71.4%
YOLOv8	GhostNet	80.8%	60.9%	73.3%
YOLOv8	proposed model	84.6%	68.6%	79.4%

Where, the numbers with frames are optimal.

According to the results in Table 3, we can see that, compared with other mainstream backbone models, the precision rate, recall rate, and mAP of the model proposed in this article can reach 84.6%, 68.6%, and 79.4%, respectively, an improvement compared to MobilenetV3-Large. Compared to ShuffleNetV2, they increased by 7.3%, 13.5%, and 11.2%, respectively; compared to GhostNet, they increased by 4.7%, 8.5%, and 8.3%, respectively.

In one study, a dependable model based on RGB-depth (RGB-D) cameras was used to accurately and automatically detect and locate the fruit-bearing branches of multiple litchi clusters simultaneously. The results showed that the detection accuracy of a litchi fruit-bearing branch was 83.33%, and positioning accuracy was 17.29° ± 24.57°. A depth camera can obtain the distance information from the object to the camera, which alleviates the impact of lighting changes on litchi detection [9]. However, compared to ordinary RGB cameras, it is difficult to apply in high-resolution litchi recognition scenarios. In order to highlight the advantages of the model in this article, a litchi scene with complex lighting was randomly selected for a comparative analysis. As shown in the figure, it is obvious that the central part of the picture is brighter and the sides are darker. Fragments from three images were selected, including Figure 7b,c (blue and yellow areas). According to the detection results, the method in this article demonstrates detection improvements in complex lighting scenes.

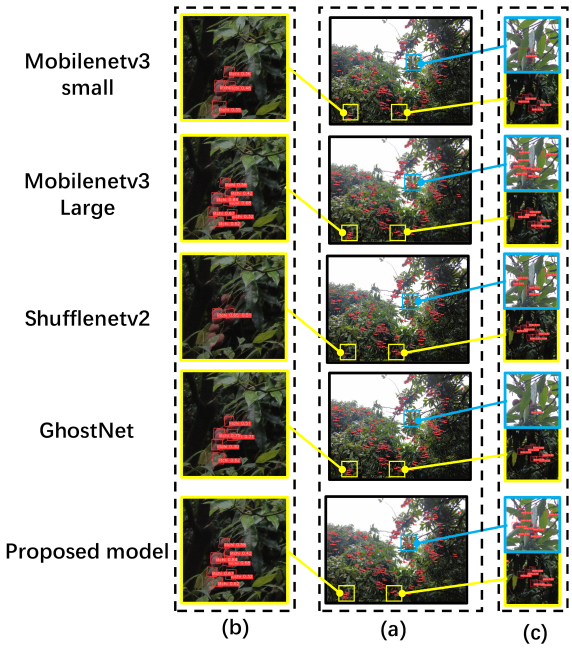

Figure 7. Performance comparison of different models. (**a**) Main picture (**b**) Partial view-1. (**c**) Partial view-2

In the occluded scene (Figure 7c blue frame area), when part of the litchi is blocked by, e.g., leaves, the model with the attention mechanism shows a better performance,

because the attention mechanism improves the litchi feature information and the extraction performance.

Figure 8 shows the detection effects of the model proposed in this article in different scenarios, including sunny, rainy, and cloudy days. It can be concluded from the detection results that the proposed model in this article has better performance in different conditions.

In addition, comparing the detection results of the other four models on this article's litchi dataset, the results based on MobileNetV3-Small contain a large number of missed detections. Due to the sacrifice of detection accuracy and the pursuit of lightweight models, GhostNet and shuffleNetV2 generally result in misdetections under occlusion. It should be noted that, when the fruits and leaves are blocked at the same time, all models result in misdetections. Due to the NMS selection problem, adjacent litchi are easily removed because they are regarded as redundant frames.

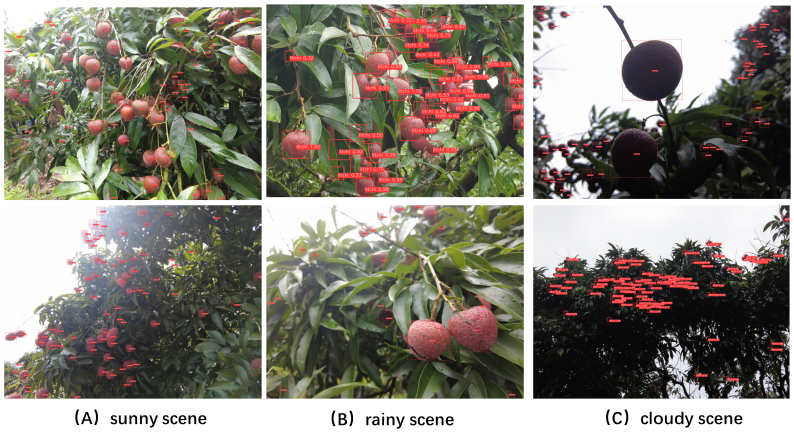

Figure 8. Performance of different scenes. (**a**) sunny, (**b**) rainy, (**c**) cloudy.

4.5. Class Activation Graph Analysis

Grad-CAM is a traditional way to interpret CNNs [42]. In this method, the importance of a convolutional neural network's feature map is determined for the output category, and the results are visualized as a heat map to explain the network's predictions for the input image. This helps explain the inner workings of convolutional neural networks, thereby improving the network's interpretability and reliability. Figure 9 shows YOLOv8 and the improved class activation map. The darker the red, the larger the value. The larger the value, the more effective the feature is, indicating that the corresponding area of the original image has a higher response and greater contribution to the network, and the more important the feature is for network prediction results. It can be seen that the improved network's receptive field is increased and its red range is more accurate, which further proves its high accuracy in extracting effective features.

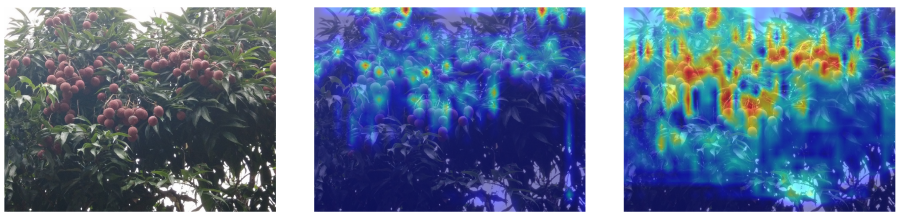

Figure 9. Comparison of visualization feature maps. (**A**) Original litchi image. (**B**) Feature map of the YOLOv8n model. (**C**) Feature map of the proposed model.

5. Conclusions

This paper proposes a litchi detection algorithm based on YOLOv8 and rotating boxes to solve the problems related to the large amounts of background and overlap

with adjacent frames introduced by traditional horizontal bounding boxes, small-target detection, etc., which reduce the accuracy of litchi detection. Based on the traditional YOLOv8n model, a transformer module is introduced after the C2f module of the eighth layer of the backbone network, and an ECA attention module is added to the neck to improve the feature extraction capability of the backbone network, which is then combined with a 160 × 160 scale detection head to enhance small-target detection. The test results showed that, compared to YOLOv8n, the model proposed in this article improved the precision, recall, and mAP by 11.7%, 5.4%, and 7.3%, respectively. At the same time, four state-of-the-art mainstream detection backbone networks were studied to compare their litchi detection performance, namely, Mobilenetv3-small, Mobilenetv3-large, Shufflenetv2, and GhostNet. The model proposed in this article exhibited a better performance, proving its utility. The litchi detection algorithm has the potential to be applied in actual scenarios and is an efficient and high-performance solution to dense litchi detection problems. It can also be applied to other fields, such as the detection of palm fruit, the detection of rice or maize tassels, and the detection of male and female flowers.

In addition, when fruits and leaves are occluded at the same time, due to the NMS selection problem, adjacent litchi are easily removed because they are regarded as redundant frames. This problem is a future research direction for litchi detection in natural scenes.

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

Funding: This research was funded by Guangdong Province Rural Revitalization Strategic Project "Orchard Agricultural Machinery and Agronomy Integration and Production Management Informatization" (2023-TS-2-4) and The Youth Tutorial Program of Guangdong Academy of Agricultural Sciences (R2021QD-023).

Data Availability Statement: No new data were created or analyzed in this study. Data sharing is not applicable to this article.

Acknowledgments: We thank Ultralytics for developing the YOLOV5 architecture (https://github.com/ultralytics/ultralytics, accessed on 28 November 2023), from which the code was taken and built upon for this project, and the Pytorch framework (https://pytorch.org/, accessed on 28 November 2023) which was adopted to train the deep learning models.

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

Abbreviations

The following abbreviations are used in this manuscript:

YOLOV8n	You Look Only Once Version eight -n scales
Otsu	maximum inter-class variance method
LDA	Linear discriminant analysis
C2f	Cross Stage Partial Network Bottleneck with Two Convolutions
C3	Cross Stage Partial Network Bottleneck with Three Convolutions
ECA	Spatial Pyramid Pooling - Fast
SVM	Support vector machines
SPPF	Spatial Pyramid Pooling - Fast
Faster R-CNN	faster region-based convolutional network
SSD	Single Shot Multibox Detector
CBAM	Convolutional Block Attention Module
HHBs	Horizontal bounding boxes
OBBs	Oriented bounding boxes
ECA	Efficient Channel Attention Network
MLP	multilayer perceptron
FPN	Feature pyramid networks

CBS	Convolutional Bottleneck with SiLU
NMS	non-maximum suppression
NLP	Neuro-Linguistic Programming
MHA	Multi-head attention
BN	Batch normalization
ChatGPT	Chatbot program Chat Generative Pre-trained Transformer
SiLU	Sigmoid Linear Unit
IOU	Intersection over Union
CIOU	Complete intersection over Union
mAP	Mean average precision
DFL	Distribution Focal Loss

References

1. Chen, X.; Wang, W.; Huang, C.; Wang, Y.; Fu, H.; Li, J. Study of the Group Vibrational Detachment Characteristics of Litchi (Litchi chinensis Sonn) Clusters. *Agriculture* **2023**, *13*, 1065. [CrossRef]
2. Xiong, J.; He, Z.; Lin, R.; Liu, Z.; Bu, R.; Yang, Z.; Peng, H.; Zou, X. Visual positioning technology of picking robots for dynamic litchi clusters with disturbance. *Comput. Electron. Agric.* **2018**, *151*, 226–237. [CrossRef]
3. Lei, X.; Yuan, Q.; Xyu, T.; Qi, Y.; Zeng, J.; Huang, K.; Sun, Y.; Herbst, A.; Lyu, X. Technologies and Equipment of Mechanized Blossom Thinning in Orchards: A Review. *Agronomy* **2023**, *13*, 2753. [CrossRef]
4. Xiong, Z.; Wang, L.; Zhao, Y.; Lan, Y. Precision Detection of Dense Litchi Fruit in UAV Images Based on Improved YOLOv5 Model. *Remote. Sens.* **2023**, *15*, 4017. [CrossRef]
5. Xiong, J.; Lin, R.; Liu, Z.; He, Z.; Tang, L.; Yang, Z.; Zou, X. The recognition of litchi clusters and the calculation of picking point in a nocturnal natural environment. *Biosyst. Eng.* **2018**, *166*, 44–57. [CrossRef]
6. Xiong, J.; Lin, R.; Bu, R.; Liu, Z.; Yang, Z.; Yu, L. A Micro-Damage Detection Method of Litchi Fruit Using Hyperspectral Imaging Technology. *Sensors* **2018**, *18*, 700. [CrossRef]
7. Wu, J.; Zhang, S.; Zou, T.; Dong, L.; Peng, Z.; Wang, H. A Dense Litchi Target Recognition Algorithm for Large Scenes. *Math. Probl. Eng.* **2022**, *2022*, 4648105. [CrossRef]
8. Jiao, Z.; Huang, K.; Jia, G.; Lei, H.; Cai, Y.; Zhong, Z. An effective litchi detection method based on edge devices in a complex scene. *Biosyst. Eng.* **2022**, *222*, 15–28. [CrossRef]
9. Li, J.; Tang, Y.; Zou, X.; Lin, G.; Wang, H. Detection of Fruit-Bearing Branches and Localization of Litchi Clusters for Vision-Based Harvesting Robots. *IEEE Access* **2020**, *8*, 117746–117758. [CrossRef]
10. Liang, C.; Xiong, J.; Zheng, Z.; Zhong, Z.; Li, Z.; Chen, S.; Yang, Z. A visual detection method for nighttime litchi fruits and fruiting stems. *Comput. Electron. Agric.* **2020**, *169*, 105192. [CrossRef]
11. He, Z.L.; Xiong, J.T.; Lin, R.; Zou, X.; Tang, L.Y.; Yang, Z.G.; Liu, Z.; Song, G. A method of green litchi recognition in natural environment based on improved LDA classifier. *Comput. Electron. Agric.* **2017**, *140*, 159–167. [CrossRef]
12. Guo, Q.; Chen, Y.; Tang, Y.; Zhuang, J.; He, Y.; Hou, C.; Chu, X.; Zhong, Z.; Luo, S. Lychee Fruit Detection Based on Monocular Machine Vision in Orchard Environment. *Sensors* **2019**, *19*, 4091. [CrossRef] [PubMed]
13. Yu, L.; Xiong, J.; Fang, X.; Yang, Z.; Chen, Y.; Lin, X.; Chen, S. A litchi fruit recognition method in a natural environment using RGB-D images. *Biosyst. Eng.* **2021**, *204*, 50–63. [CrossRef]
14. Ortiz, C.; Torregrosa, A.; Castro-García, S. Citrus Fruit Movement Assessment Related to Fruit Damage during Harvesting with an Experimental Low-Frequency–High-Amplitude Device. *Agronomy* **2022**, *12*, 1337. [CrossRef]
15. Mark, E.; De Kleine, M.K. A Semi-Automated Harvesting Prototype for Shaking Fruit Tree Limbs. *Trans. ASABE* **2015**, *58*, 1461–1470. [CrossRef]
16. Torregrosa, A.; Albert, F.; Aleixos, N.; Ortiz, C.; Blasco, J. Analysis of the detachment of citrus fruits by vibration using artificial vision. *Biosyst. Eng.* **2014**, *119*, 1–12. [CrossRef]
17. Bu, L.; Hu, G.; Chen, C.; Sugirbay, A.; Chen, J. Experimental and simulation analysis of optimum picking patterns for robotic apple harvesting. *Sci. Hortic.* **2019**, *261*, 108937. [CrossRef]
18. Li, T.; Sun, M.; He, Q.; Zhang, G.; Shi, G.; Ding, X.; Lin, S. Tomato recognition and location algorithm based on improved YOLOv5. *Comput. Electron. Agric.* **2023**, *208*, 107759. [CrossRef]
19. Han, C.; Wu, W.; Luo, X.; Li, J. Visual Navigation and Obstacle Avoidance Control for Agricultural Robots via LiDAR and Camera. *Remote. Sens.* **2023**, *15*, 5402. [CrossRef]
20. Wang, H.; Dong, L.; Zhou, H.; Luo, L.; Lin, G.; Wu, J.; Tang, Y. YOLOv3-Litchi Detection Method of Densely Distributed Litchi in Large Vision Scenes. *Math. Probl. Eng.* **2021**, *2021*, 8883015. [CrossRef]
21. Xie, J.; Peng, J.; Wang, J.; Chen, B.; Jing, T.; Sun, D.; Gao, P.; Wang, W.; Lu, J.; Yetan, R.; et al. Litchi Detection in a Complex Natural Environment Using the YOLOv5-Litchi Model. *Agronomy* **2022**, *12*, 3054. [CrossRef]
22. Wang, L.; Zhao, Y.; Xiong, Z.; Wang, S.; Li, Y.; Lan, Y. Fast and precise detection of litchi fruits for yield estimation based on the improved YOLOv5 model. *Front. Plant Sci.* **2022**, *13*, 965425. [CrossRef] [PubMed]

23. Qi, X.; Dong, J.; Lan, Y.; Zhu, H. Method for Identifying Litchi Picking Position Based on YOLOv5 and PSPNet. *Remote. Sens.* **2022**, *14*, 2004. [CrossRef]
24. Wang, C.; Zou, X.; Tang, Y.; Luo, L.; Feng, W. Localisation of litchi in an unstructured environment using binocular stereo vision. *Biosyst. Eng.* **2016**, *145*, 39–51. [CrossRef]
25. Peng, H.; Xue, C.; Shao, Y.; Chen, K.; Liu, H.; Xiong, J.; Chen, H.; Gao, Z.; Yang, Z. Litchi detection in the field using an improved YOLOv3 model. *Int. J. Agric. Biol. Eng.* **2022**, *15*, 211–220. [CrossRef]
26. Wang, H.; Lin, Y.; Xu, X.; Chen, Z.; Wu, Z.; Tang, Y. A Study on Long-Close Distance Coordination Control Strategy for Litchi Picking. *Agronomy* **2022**, *12*, 1520. [CrossRef]
27. Xie, J.; Zhang, X.; Liu, Z.; Liao, F.; Wang, W.; Li, J. Detection of Litchi Leaf Diseases and Insect Pests Based on Improved FCOS. *Agronomy* **2023**, *13*, 1314. [CrossRef]
28. Wen, L.; Cheng, Y.; Fang, Y.; Li, X. A comprehensive survey of oriented object detection in remote sensing images. *Expert Syst. Appl.* **2023**, *224*, 119960. [CrossRef]
29. Yang, W.; Wu, J.; Zhang, J.; Gao, K.; Du, R.; Wu, Z.; Firkat, E.; Li, D. Deformable convolution and coordinate attention for fast cattle detection. *Comput. Electron. Agric.* **2023**, *211*, 108006. [CrossRef]
30. Yang, H.; Shi, Y.; Wang, X. Detection Method of Fry Feeding Status Based on YOLO Lightweight Network by Shallow Underwater Images. *Electronics* **2022**, *11*, 3856. [CrossRef]
31. 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. [CrossRef]
32. Mekhalfi, M.L.; Nicolo, C.; Bazi, Y.; Rahhal, M.M.A.; Alsharif, N.A.; Maghayreh, E.A. Contrasting YOLOv5, Transformer, and EfficientDet Detectors for Crop Circle Detection in Desert. *IEEE Geosci. Remote. Sens. Lett.* **2021**, *19*, 1–5. [CrossRef]
33. Roy, A.M.; Bhaduri, J. DenseSPH-YOLOv5: An automated damage detection model based on DenseNet and Swin-Transformer prediction head-enabled YOLOv5 with attention mechanism. *Adv. Eng. Inform.* **2023**, *56*, 102007. [CrossRef]
34. Guo, Z.; Wang, C.; Yang, G.; Huang, Z.; Li, G. MSFT-YOLO: Improved YOLOv5 Based on Transformer for Detecting Defects of Steel Surface. *Sensors* **2022**, *22*, 3467. [CrossRef] [PubMed]
35. Tang, Z.; Lu, J.; Chen, Z.; Qi, F.; Zhang, L. Improved Pest-YOLO: Real-time pest detection based on efficient channel attention mechanism and transformer encoder. *Ecol. Inform.* **2023**, *78*, 102340. [CrossRef]
36. Xia, X.; Chai, X.; Li, Z.; Zhang, N.; Sun, T. MTYOLOX: Multi-transformers-enabled YOLO for tree-level apple inflorescences detection and density mapping. *Comput. Electron. Agric.* **2023**, *209*, 107803. [CrossRef]
37. Li, Y.; Miao, N.; Ma, L.; Shuang, F.; Huang, X. Transformer for object detection: Review and benchmark. *Eng. Appl. Artif. Intell.* **2023**, *126*, 107021. [CrossRef]
38. Zhu, D.; Wang, D. Transformers and their application to medical image processing: A review. *J. Radiat. Res. Appl. Sci.* **2023**, *16*, 100680. [CrossRef]
39. Howard, A.; Sandler, M.; Chu, G.; Chen, L.C.; Chen, B.; Tan, M.; Wang, W.; Zhu, Y.; Pang, R.; Vasudevan, V.; et al. Searching for MobileNetV3. *arXiv* **2019**, arXiv:1905.02244.
40. Zhu, X.; Lyu, S.; Wang, X.; Zhao, Q. TPH-YOLOv5: Improved YOLOv5 Based on Transformer Prediction Head for Object Detection on Drone-captured Scenarios. In Proceedings of the 2021 IEEE/CVF International Conference on Computer Vision Workshops (ICCVW), Montreal, BC, Canada, 11–17 October 2021; pp. 2778–2788. [CrossRef]
41. Li, X.; Hu, X.; Yang, J. Spatial Group-wise Enhance: Improving Semantic Feature Learning in Convolutional Networks. *arXiv* **2019**, arXiv:1905.09646.
42. Ru, C.; Zhang, S.; Qu, C.; Zhang, Z. The High-Precision Detection Method for Insulators' Self-Explosion Defect Based on the Unmanned Aerial Vehicle with Improved Lightweight ECA-YOLOX-Tiny Model. *Appl. Sci.* **2022**, *12*, 9314. [CrossRef]
43. Gao, C.; Tang, T.; Wu, W.; Zhang, F.; Luo, Y.; Wu, W.; Yao, B.; Li, J. Hyperspectral Prediction Model of Nitrogen Content in Citrus Leaves Based on the CEEMDAN–SR Algorithm. *Remote. Sens.* **2023**, *15*, 5013. [CrossRef]
44. Su, Y.; Liu, Q.; Xie, W.; Hu, P. YOLO-LOGO: A transformer-based YOLO segmentation model for breast mass detection and segmentation in digital mammograms. *Comput. Methods Programs Biomed.* **2022**, *221*, 106903. [CrossRef] [PubMed]
45. Peng, H.; Zhong, J.; Liu, H.; Li, J.; Yao, M.; Zhang, X. ResDense-focal-DeepLabV3+ enabled litchi branch semantic segmentation for robotic harvesting. *Comput. Electron. Agric.* **2023**, *206*, 107691. [CrossRef]
46. Wang, M.; Yang, B.; Wang, X.; Yang, C.; Xu, J.; Mu, B.; Xiong, K.; Li, Y. YOLO-T: Multitarget Intelligent Recognition Method for X-ray Images Based on the YOLO and Transformer Models. *Appl. Sci.* **2022**, *12*, 11848. [CrossRef]
47. Han, K.; Wang, Y.; Tian, Q.; Guo, J.; Xu, C.; Xu, C. GhostNet: More Features from Cheap Operations. *arXiv* **2019**, arXiv:1911.11907.