forests

MDPI

Article

YOLOv5-ACS: Improved Model for Apple Detection and Positioning in Apple Forests in Complex Scenes

Jianping Liu [1,2], Chenyang Wang [1,*] and Jialu Xing [1]

[1] College of Computer Science and Engineering, North Minzu University, Yinchuan 750021, China; liujianping01@nmu.edu.cn (J.L.); 20227440@stu.nmu.edu.cn (J.X.)
[2] The Key Laboratory of Images and Graphics Intelligent Processing of State Ethnic Affairs Commission, North Minzu University, Yinchuan 750021, China
* Correspondence: 20227511@stu.nmu.edu.cn

Abstract: Apple orchards, as an important center of economic activity in forestry special crops, can achieve yield prediction and automated harvesting by detecting and locating apples. Small apples, occlusion, dim lighting at night, blurriness, cluttered backgrounds, and other complex scenes significantly affect the automatic harvesting and yield estimation of apples. To address these issues, this study proposes an apple detection algorithm, "YOLOv5-ACS (Apple in Complex Scenes)", based on YOLOv5s. Firstly, the space-to-depth-conv module is introduced to avoid information loss, and a squeeze-and-excitation block is added in C3 to learn more important information. Secondly, the context augmentation module is incorporated to enrich the context information of the feature pyramid network. By combining the shallow features of the backbone P2, the low-level features of the object are retained. Finally, the addition of the context aggregation block and CoordConv aggregates the spatial context pixel by pixel, perceives the spatial information of the feature map, and enhances the semantic information and global perceptual ability of the object. We conducted comparative tests in various complex scenarios and validated the robustness of YOLOv5-ACS. The method achieved 98.3% and 74.3% for mAP@0.5 and mAP@0.5:0.95, respectively, demonstrating excellent detection capabilities. This paper creates a complex scene dataset of apples on trees and designs an improved model, which can provide accurate recognition and positioning for automatic harvesting robots to improve production efficiency.

Keywords: apple trees; apple detection; smart farming; deep learning; context aggregation; attention mechanisms; CoordConv

check for updates

Citation: Liu, J.; Wang, C.; Xing, J. YOLOv5-ACS: Improved Model for Apple Detection and Positioning in Apple Forests in Complex Scenes. *Forests* **2023**, *14*, 2304. https://doi.org/10.3390/f14122304

Academic Editor: Asunción Cámara-Obregón

Received: 12 October 2023
Revised: 17 November 2023
Accepted: 22 November 2023
Published: 24 November 2023

Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

Driven by the continuous growth of global demand, apple production has significantly increased, reaching 75 million tons in 2020/21 [1]. Due to the complexity of the modern apple orchard environment, fruit harvesting continues to rely on manual labor. However, traditional manual harvesting methods suffer from challenges such as high labor intensity, long work cycles, low efficiency, and difficulties in nighttime harvesting. Apple harvesting is a seasonal endeavor, and the supply of seasonal labor is often unstable. Automation of the harvesting process is not subject to time constraints and can significantly improve the harvesting efficiency and yield stability. Furthermore, environmental factors have minimal impact on harvesting machinery, enabling it to perform harvesting tasks efficiently and continuously. This is particularly advantageous in nighttime environments where harvesting robots can not only mitigate the fatigue and errors associated with manual labor, but also ensure timely fruit collection, reducing fruit losses and waste. Apple-harvesting robot technology is a critical factor in improving the efficiency and quality of apple production and addressing labor shortages in orchards. It holds paramount importance in reducing labor costs, alleviating labor shortages in orchards, and enhancing the economic benefits for fruit farmers [2].

Guided by intelligent technology, smart farming promotes the automation of farming production, such as automatic irrigation, fertilization, and harvesting, greatly reducing the labor intensity of farmers. Identifying and pinpointing crops using object detection techniques is a crucial step in achieving harvest automation. Traditional object detection algorithms primarily rely on machine learning techniques, employing feature extraction and classifiers for classification and localization. However, as deep learning gains momentum, it is gradually displacing conventional methods. The object detection algorithm based on deep learning can directly learn feature representations with object recognition capabilities from the data [3], removing the requirement for feature extraction by hand. When dealing with complex scenes and diverse objects, deep learning methods demonstrate superior accuracy and performance.

The domain of apple detection has witnessed substantial progress, which is largely attributed to advancements in object detection algorithms.

Chu et al. [4] proposed an occlusion-aware network to detect occluded apples in densely arranged clusters by introducing a feature extension structure to extract additional features of the occluded apples, achieving an accuracy of 94%. Sun et al. [5] embedded an extended layer of ResNet50 conv1 in the lowest level of the feature pyramid network for immature small apples, and designed a decoupled aggregation module to supplement spatial positioning information, improving the detection effect for small apples. Xuan et al. [6] achieved a lightweight version of YOLOv3 by removing the fully connected layer in the feature extraction network and the eight-fold down-sampling detection branch in the neck network. Under normal lighting conditions, the F1 scores for red and green apples reached 95.0% and 94.0%, respectively. Meng [7] proposed an optimized deformable DETR model to improve the detection accuracy of green fruits and small targets. They use the ResNeXt network as the backbone of the model and introduce a deformable attention mechanism to fuse multi-scale features, the detection accuracy of AP_{50} and AP_s was 80.4% and 35.4%, respectively. Although the aforementioned improved models perform well, they are tailored to specific scenarios. In actual apple production management, it is possible to encounter all of the aforementioned scenarios simultaneously. Apart from the mentioned scenarios, apple detection is influenced by other factors, such as differences in image acquisition device resolutions and the need to work at night in emergencies, where nighttime lighting may be dim or uneven. These factors severely impact the effectiveness of apple detection in practical, complex scenes. Therefore, achieving efficient and stable apple detection in complex modern orchard scenes is of paramount importance for realizing apple-harvesting automation. In this paper, nighttime scene data were incorporated. Considering the complex orchard environment and the high efficiency required during harvesting, we selected YOLOv5s as the base model and proposed a YOLOv5s-ACS model suitable for working in practical scenarios.

Research Motivation and Contribution

(1) Aiming at the problem of apples on the tree: occlusion, small apples, complex background information, and the effect of light, we collected and processed the relevant datasets. Subsequently, we constructed a complex scene dataset for object detection of apples on the tree. (2) To address the aforementioned complex scenarios, corresponding solutions are proposed. First, due to the inclusion of low-resolution and small apples in the dataset, SPD-Conv (space-to-depth-conv) and C3SE (squeeze-and-excitation) were incorporated into the backbone, retaining discriminative feature information and focusing on more important channel information. Secondly, to provide feature pyramid networks (FPN) with richer information so that they can better detect small apples and night apples, a CAM (context augmentation module) was used to replace the spatial pyramid pooling-fast (SPPF) module, and the backbone's P2 shallow features were introduced into the neck network, retaining shallow features and paying attention to detailed information while a fusion of different receptive fields enrich the semantic information. Finally, the CABlock (context aggregation block) and CoordConv were added in front of the detection

head to filter useless background information, perceive spatial information, and improve the detection ability of occluded targets. (3) YOLOv5-ACS achieves optimal results on various complex scenes, both original and enhanced. (4) The proposed YOLOv5-ACS can adapt to various complex scenarios encountered during the production process. It can consistently deliver precise identification and positioning for automated harvesting, even during nighttime harvesting in emergencies.

2. Related Work

2.1. Object Detection

Object detection, which has received much attention, aims to detect and recognize all the salient objects in the whole image [8]. Due to the improvement in the computing power of computer hardware, significant advances have been achieved in object detection technology [9]. Object detection can be roughly categorized into three types (as shown in Figure 1): (1) Two-stage object detection algorithms based on region proposals, such as the R-CNN series (R-CNN [10], Fast R-CNN [11], and Faster R-CNN [12]) and SPPNet [13]. (2) One-stage object detection algorithms based on integrated convolutional networks, such as the YOLO [14] series, SSD [15], and RetinaNet [16]. (3) Object detection algorithms based on transformers, such as the DETR [17] (detection transformer) and deformable DETR [18]. The two-stage target detection algorithm mainly includes two steps. First, it extracts object regions, i.e., it extracts candidate bounding boxes based on the image. Second, it performs CNN-based classification and recognition on these regions, refining the detection results based on candidate regions [19]. While two-stage object detection achieves a high detection accuracy, it is relatively slower in terms of detection speed. A better trade-off between speed and accuracy is achieved by one-stage object detection, which converts the object detection issue into the regression problem of locating object boxes and calculating class probabilities [20]. Neither one-stage nor two-stage object detection methods effectively utilize the attention mechanism. Addressing this issue, the DETR introduced the transformer architecture into the field of object detection. The DETR does not require anchors or non-maximum suppression (NMS). Rather, it uses object queries for feature interaction and approaches object detection as a set prediction problem [21]. However, the transformer model is large and computationally intensive, which makes it less suitable for deployment and practical application in resource-constrained environments. Given the complex scenes and detection efficiency of apple detection, the one-stage object detection approach aligns better with the objectives of this paper.

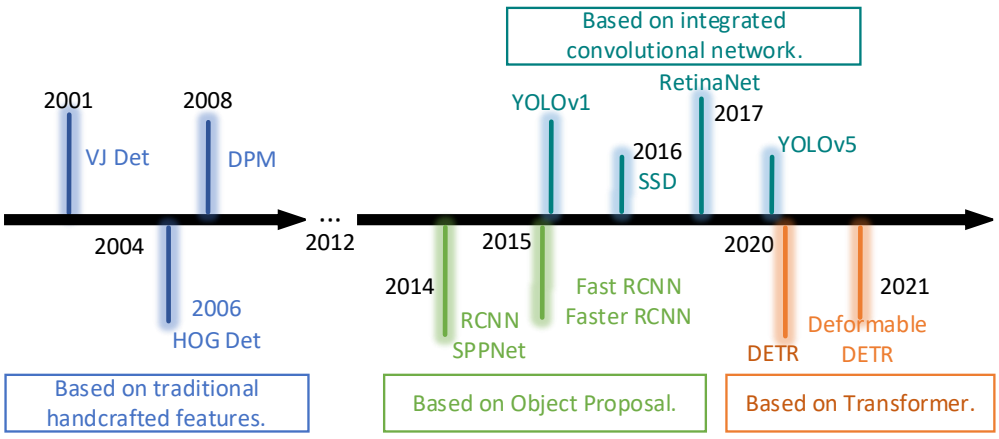


Figure 1. Development history of object detection network.

2.2. YOLOv5 Network

The one-stage object detection model, YOLOv5, was proposed in 2020 and has achieved widespread adoption in the field of smart farming [22]. The four main parts of the deep learning YOLOv5 model are the input, backbone, neck, and head.

Adaptive image scaling, adaptive anchor computation, and mosaic image augmentation are included in the input stage [23]. Data augmentation is employed to avoid overfitting due to insufficient data. Appropriate anchor boxes are calculated using adaptive anchor boxes. Adaptive image scaling is implemented to reduce model computation and enhance the detection capability for small targets [24].

There are three main modules in the backbone: CBS (Conv + BatchNorm + SiLU), C3, and SPPF. They are mainly responsible for gradually extracting different features from the image from low levels to high levels. The C3 module divides the feature maps of a stage into two parts, employing a split and merge strategy that is applied across stages [25]. It reduces the probability of redundant information integration and minimizes the repetition of gradient information. As a result, the YOLOv5 network exhibits an enhanced learning capability and experiences reduced inference computations.

The neck network, utilizes the FPN + PAN (path aggregation network) structure. After the feature maps of different scales are converted to the same scale, direct cascade technology is applied to merge the features [26]. YOLOv5 leverages FPN-PAN to conduct dual-feature fusion, integrating information from various scales. This enables the image features extracted by the model to be more comprehensive.

The head network performs the final regression predictions. Performing recognition and localization through detection heads at different scales.

YOLOv5 has excellent scalability and excellent detection accuracy, which can meet the needs of practical applications, and it has been widely deployed in production environments [27]. Users can scale YOLOv5 according to their task requirements and computational resources, ranging from lightweight networks like YOLOv5s to larger networks like YOLOv5x. YOLOv5 allows users to add custom layers and loss functions based on different task demands. Additionally, YOLOv5 offers numerous plug-and-play modules, such as attention modules (SE, CBAM, and CA), and various loss functions suitable for different tasks (EIOU loss, and WIOU loss).

2.3. Application of Object Detection in Fruit Detection on the Tree

The detection of fruit on the tree presents similar challenges, such as different ripeness, different sizes, occlusion, and high densities. Thanks to the rapid development of deep learning, the application of object detection in fruit detection on trees, such as apples, oranges, pears, kiwis, and more, has become extremely widespread. Mu et al. [28] proposed an object recognition algorithm based on an the improved AlexNet to identify occluded kiwi fruits in complex environments, which improves the detection accuracy for cases that involve leaf occlusion or overlapping fruit overlap, for example. Li et al. [29] combined the growth patterns of Orah mandarin oranges in a natural setting and used MobileNet_v1 as the feature extraction network for the SSD model, improving the model's detection performance on small targets. Wu et al. [30] introduced a Light-YOLOv3 algorithm designed to swiftly detect apples in complex backgrounds. This algorithm uses a feature extraction network made up of concatenated homogeneous residual blocks to simplify the feature map scale of object detection. Li et al. [31] proposed a multi-scale collaborative perception network called YOLOv5s-FP for detecting pears, which combines specific and holistic features to address issues such as occlusion in orchards and the diversity of image capture positions. The aforementioned advancements have facilitated the progress of tree fruit detection and can serve as a guide for detecting comparable fruits on trees.

The fast and precise detection and localization of fruits on the tree are vital for automating harvesting, estimating yield, and intelligently managing orchards. They represent a significant step towards achieving smart farming.

3. Materials and Methods

3.1. Data Collection

The apple image dataset consists of two parts: daytime and nighttime, both of which were used for model training and validation, as shown in Table 1. The daytime apple image

dataset, consisting of 800 apple images with a resolution of 1920 × 1080, was downloaded from the GitHub repository. The nighttime apple image dataset, captured under artificial lighting conditions, was downloaded from the WSU Research Exchange, and contains 200 apple images with a resolution of 1280 × 960. The example images of apples in different scenarios are shown in Figure 2.

Table 1. Apple dataset.

Dataset	Number	Resolution Ratio	Feature and Manual Enhancement
Daytime	800	1920 × 1080	Daytime light: sunlight, backlight; Night light: dim light (76), uneven artificial light (124); Occlusion: leaves occlusion, branch occlusion, occlusion between apples; Individual difference: size difference, maturity differences; Overall: high density arrangement;
Nighttime	200	1280 × 960	The overall brightening (800) and darkening (800) of daytime images; Vertical blur (1000) and horizontal blur (1000)

(**a**) High density, Occlusion (**b**) Maturity differences (**c**) Sunlight

(**d**) Daytime motion blur (**e**) Dimming (**f**) Brightening

(**g**) Artificial light (**h**) Dim light (**i**) Nighttime motion blur

Figure 2. Apple samples from different scenes.

The apple image dataset was annotated using LabelImg (Version 1.8.6). Bounding boxes were used to label the targets in the apple images, and the annotations were saved as txt files. Each file contains the category ID and coordinate information of the target.

Data augmentation techniques were utilized to increase the sample diversity while preventing overfitting caused by insufficient data. To expand the daytime dataset, we applied brightness augmentation and brightness reduction. OpenCV was employed to introduce blur effects to the image dataset, using a specific type of motion blur that provided directional blurring effects for both daytime and nighttime images. Two effects were utilized: vertical blur and horizontal blur. In total, there were 4600 images after the augmentation. We randomly selected 80% (3680 images) of all images as the training set, and the remainder (920 images) as the validation set.

3.2. Apple Object Detection Based on Improved YOLOv5s Network

YOLOv5 demonstrates significant advantages in both detection accuracy and speed [32]. Its simplified version, YOLOv5s, with a lower model complexity, is suitable for running and implementing on mobile devices [33]. The overall architecture of the YOLOv5-ACS for object detection of apples on the tree in complex orchard environments is illustrated in Figure 3. In this study, the SPD module was added before each C3 module of YOLOv5s, and the subsequent Conv module was changed to have a stride of 1 in order to form an SPD-Conv module, which preserves fine-grained information of low-resolution images. The SE attention mechanism was introduced into C3 to adaptively learn the importance of each channel, obtaining more important feature information and enhancing the detection capability for small targets. The CAM was added to the end of the backbone to enhance the features. In the multi-scale feature fusion stage, P2 shallow features from the backbone were also fused and the output feature layer size of the detection head was increased to leverage more fine-grained feature information. CABlock and CoordConv were added before the detection head to reduce information confusion, generate feature maps with coordinate information, and solve the interference caused by different types of occlusion.

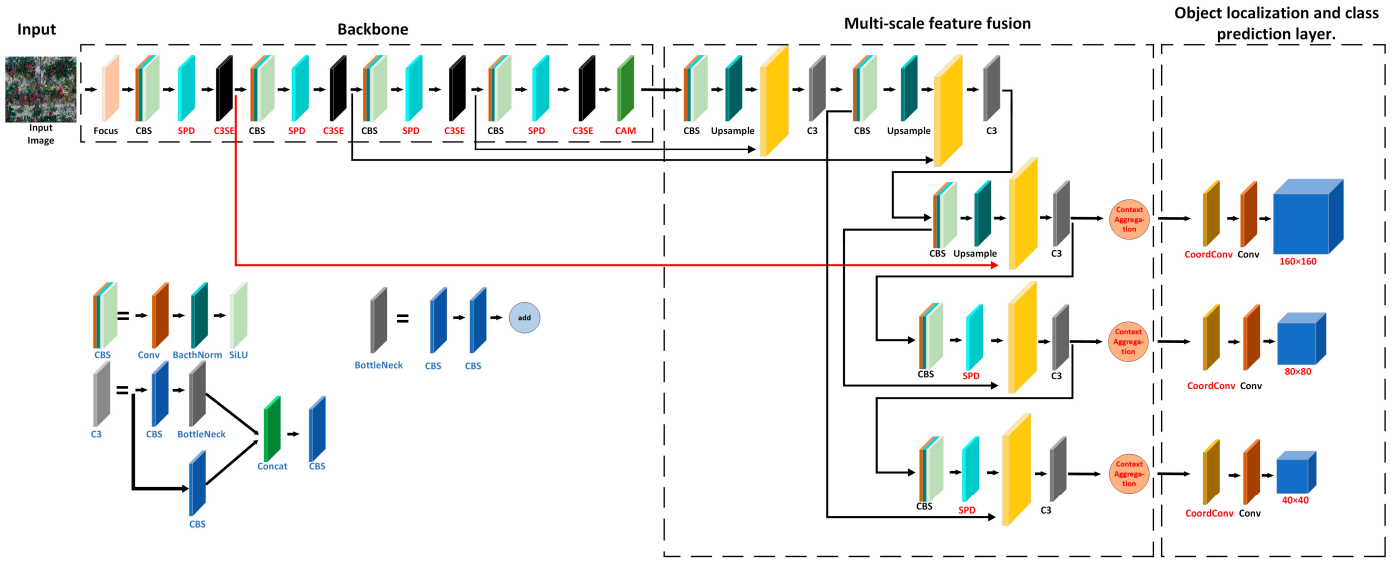


Figure 3. YOLOv5-ACS network structure. (Concat represents tensor concatenation, and SILU is the activation function).

3.2.1. Focus Module

Since apple images contain objects of different scales, with a majority of small and densely packed objects, the focus module is used to reduce information loss caused by down-sampling. The module improves the perception of small objects while reducing the false negative rate for large objects. The focus layer utilizes a slicing operation to split the images or feature maps from high-resolution into multiple low-resolution images, achieved by column-wise sampling and concatenation, as shown in Figure 4.

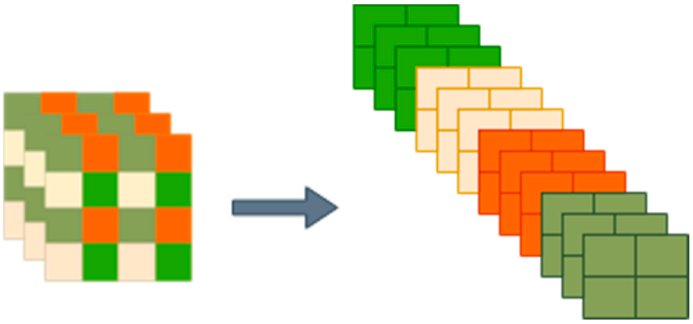


Figure 4. Focus module. (The gray arrow represents the conversion process of the data).

Initially, the image with a resolution of 640 × 640 × 3 is input to the focus structure. Through a slicing operation, it is transformed into a 320 × 320 × 12 feature map. Following this, a concatenation operation is applied, and subsequently, a convolution operation (CBS) is performed, resulting in a final feature map of size 320 × 320 × 64. This approach serves to mitigate information loss caused by down-sampling.

3.2.2. SPD-Conv Module

Convolutional neural networks (CNNs) utilize stride convolutions and/or pooling layers to filter out redundant pixel information. However, when dealing with complex scenes that contain low-resolution images and numerous small targets, this approach causes the loss of fine-grained information, leading to a decline in performance [34].

Due to the loss of fine-grained information in existing CNN architectures and the presence of low-resolution images and small objects in the complex scene dataset created in this study, a novel convolution called SPD-convolution (SPD-Conv) was introduced. The SPD-Conv consists of a space-to-depth layer and a non-strided convolution layer, aiming to avoid the use of convolutional strides and pooling layers. The SPD layer performs down-sampling on the feature map F while preserving all information in the channel dimension, thus preventing information loss.

- Space-to-Depth (SPD) Module

For any feature map F (S,S,C1), the sub-feature map sequence is obtained by dividing it in the following way:

$$\begin{gathered} f_{0,0} = F\,[0{:}S{:}scale,\ 0{:}S{:}scale] \\ f_{1,0} = F\,[1{:}S{:}scale,\ 0{:}S{:}scale] \\ \cdots \\ f_{scale\text{-}1,0} = F\,[scale\text{-}1{:}S{:}scale,\ 0{:}S{:}scale] \end{gathered} \tag{1}$$

$$\begin{gathered} f_{0,1} = F\,[0{:}S{:}scale,\ 1{:}S{:}scale] \\ f_{1,1} = F\,[1{:}S{:}scale,\ 1{:}S{:}scale] \\ \cdots \\ f_{scale\text{-}1,1} = F\,[scale\text{-}1{:}S{:}scale,\ 1{:}S{:}scale] \end{gathered} \tag{2}$$

$$\begin{gathered} f_{0,scale\text{-}1} = F\,[0{:}S{:}scale,\ scale\text{-}1{:}S{:}scale] \\ f_{1,scale\text{-}1} = F\,[1{:}S{:}scale,\ scale\text{-}1{:}S{:}scale] \\ f_{scale\text{-}1,scale\text{-}1} = F\,[scale\text{-}1{:}S{:}scale,\ scale\text{-}1{:}S{:}scale] \end{gathered} \tag{3}$$

Typically, the feature map F is down-sampled according to a factor of scale and is partitioned from the original feature map F (S,S,C1) into sub-feature maps F′ $(\frac{S}{scale}, \frac{S}{scale}, C1)$ with the number of sub-feature maps $scale^2$. Then, sub-feature maps F′ $(\frac{S}{scale}, \frac{S}{scale}, C1)$ are spliced along the channel as F″ $(\frac{S}{scale}, \frac{S}{scale}, scale^2C1)$. The model can maximize the retention of information in F through SPD operations, making the model more focused on small targets in the feature map. To preserve the information in the image as much as possible, we used a scale factor of 2, as shown in Figure 5.

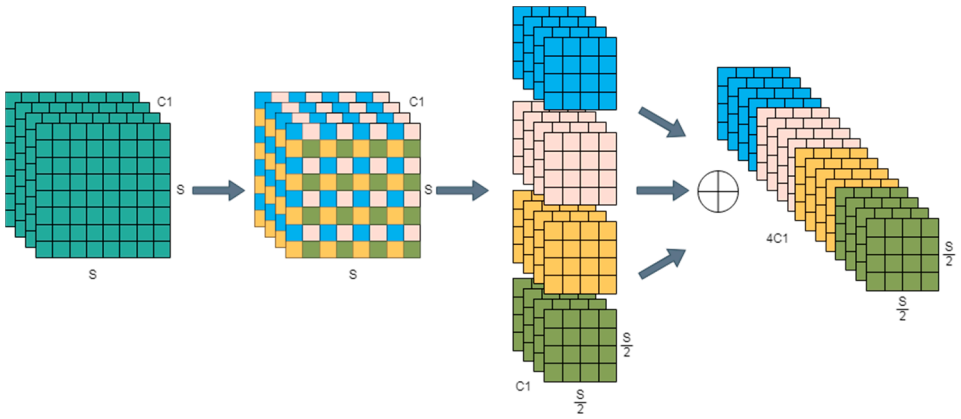


Figure 5. Space-to-depth (SPD) module. (The gray arrow represents the conversion process of the data; (S, S, C1) represents the shape of the tensor (width, height, depth); the cross icon operation is concatenated).

- Nonstrided Convolution Module

After down-sampling through the SPD module, the next transformation $F'' \left(\frac{S}{scale}, \frac{S}{scale}, scale^2C1\right) \rightarrow F''' \left(\frac{S}{scale}, \frac{S}{scale}, C2\right)$ is carried out through a convolution-free step layer (i.e., stride = 1) with a C2 filter, where $C2 < scale^2C1$, as shown in Figure 6.

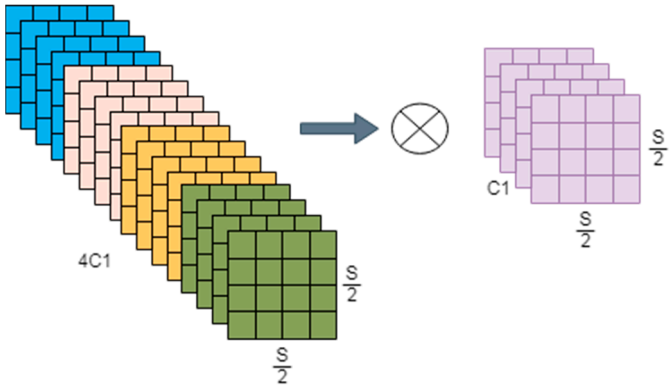


Figure 6. Non-strided convolution module. ("S" represents the width and height of the tensor shape, while "C1" represents the depth or channel dimension; the X icon represents multiply and Convolution).

3.2.3. C3SE Module

Due to the complexity of the background information in orchard environments, a significant amount of redundant information is introduced during feature extraction. C3 is a key component in YOLOv5, and therefore, the SENet block was integrated into the C3, as shown in Figure 7. The SE [35] attention mechanism assigns an attention weight to each feature channel, allowing it to focus more on the channels that are beneficial for recognizing the target, while suppressing less relevant channels. This helps to suppress irrelevant background information and improve the detection capability for small apples. Since there is a noticeable increase in channels following the SPD layer, and the SE attention mechanism operates on channel-wise attention, using the SE attention mechanism is more suitable for this scenario. In this research, we tested the effectiveness of various attention mechanisms on YOLOv5-ACS, as shown in Table 2. The experimental results confirmed that SENet is more suitable for this task, and using C3SE achieved the best average accuracy.

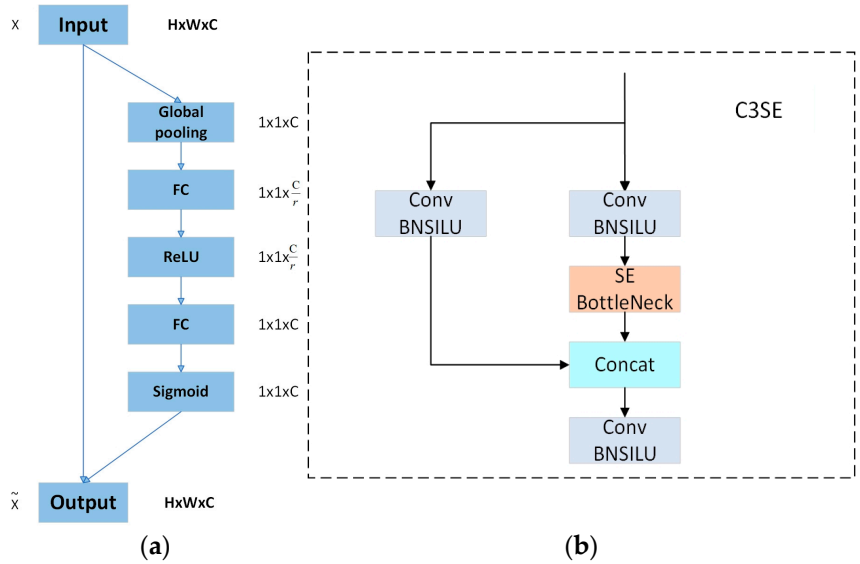


Figure 7. SENet block (**a**), C3SE module (**b**) (global pooling is the global average pooling; FC is the fully connected layer; ReLu and sigmoid represent activation functions).

Table 2. Performance of different attention mechanisms on YOLOv5-ACS.

Model	Precision (%)	Recall (%)	mAP_0.5 (%)	mAP_0.5:0.95 (%)
C3SE_YOLOv5-ACS.	95.1	93.9	98.3	74.3
C3CBAM_YOLOv5-ACS	94.6	94.0	98.2	74.0
C3ECA_YOLOv5-ACS	95.2	93.5	98.2	74.2

3.2.4. Context Augmentation Module

Due to the limitations of the network itself and the imbalance in training data, detecting small objects becomes more challenging [36]. The information about small objects gradually diminishes as the feature extraction network deepens, making the detection of small objects more reliant on contextual information. In this study, CAM was utilized to fuse features from different receptive fields to obtain contextual information. The working principle of CAM is depicted in Figure 8, where data augmentation was achieved by fusing features obtained from convolution operations with different dilated rates within the same feature map.

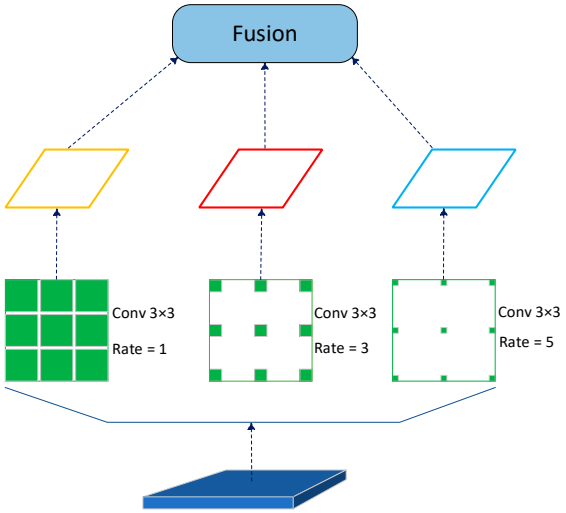


Figure 8. Context augmentation module (obtain different receptive field information using dilated convolutions with rates of 1, 3, and 5).

There are three different fusion methods, as illustrated in Figure 9: (a) weighted fusion, (b) adaptive fusion, and (c) concatenation. In methods (a) and (c), the three input feature maps are added in the channel and spatial dimensions. Method (b) is an adaptive fusion approach where spatially adaptive weights are obtained through convolution, concatenation, and Softmax.

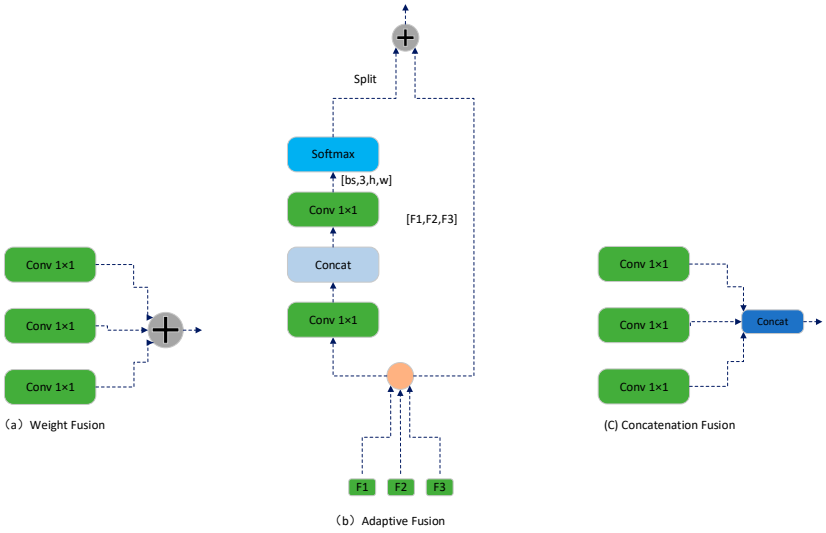


Figure 9. Ways of fusion.

The performance results of each fusion method on the improved YOLOv5s model are depicted in Table 3. The experimental findings demonstrate that the adaptive fusion method outperforms others, and thus, the YOLOv5-ACS model selected the adaptive fusion method.

Table 3. Results of different fusion methods.

Method	Precision (%)	Recall (%)	mAP_0.5 (%)	mAP_0.5:0.95 (%)
Weighted Fusion	95.1	93.9	98.3	74.1
Adaptive Fusion	95.1	93.9	98.3	74.3
Concatenation Fusion	95.1	93.6	98.2	74.1

3.2.5. Multi-Scale Feature Fusion

Apple images as inputs are usually complicated, and detecting small apples or apples in low-light conditions can be challenging since there is limited feature information available. From the perspective of receptive fields, lower-level features have smaller receptive fields, making them more localized. Smaller objects can be well preserved in the lower-level feature maps. In nighttime images, there is lower contrast and more complex backgrounds. Shallow-level features can better retain information from nighttime apple images to a certain extent.

In YOLOv5, the FPN (feature pyramid network) fuses the features from the P3, P4, and P5 layers of the backbone. However, even in the shallower P3 layer, there is still a significant loss of information for small objects. Therefore, this study additionally introduces the P2 feature map from the backbone. Additionally, the shape and size of the three output feature layers of the detection head were altered. They were increased from the original sizes of 80×80, 40×40, and 20×20 to 160×160, 80×80, and 40×40, respectively. Increasing the resolution of feature layers helps to better detect nighttime apples and small apples.

3.2.6. Context Aggregation Block

By utilizing shallow features, more background information is preserved. Therefore, we introduced the CABlock [37] to learn information per pixel and filter out unnecessary

background information, as shown in Figure 10. Within the CABlock, the pixel-wise spatial context is aggregated through

$$Q_i^j = P_i^j + a_i^j \cdot \sum_{j=1}^{N_i} \left[\frac{\exp\left(w_k P_i^j\right)}{\sum_{m=1}^{N_i} \exp(w_k P_i^m)} \cdot w_v P_i^j \right] \tag{4}$$

where P_i and Q_i refer to the input and output feature maps, respectively, from the i-th layer of the neck network. Each of these feature maps contains N_i pixels. The variables j and m are used to represent the indices of each pixel, and their values range from 1 to N_i. The variables w_k and w_v represent linear transformation matrices, which are used to project the feature maps. The a_i is used to balance the degree of aggregation of each pixel with the global spatial context. It represents a re-weighting matrix with the same shape as P_i^j and Q_i^j.

$$a_i^j = \frac{\exp\left(w_a P_i^j\right)}{\sum_{n=1}^{N_i} \exp(w_a P_i^n)} \tag{5}$$

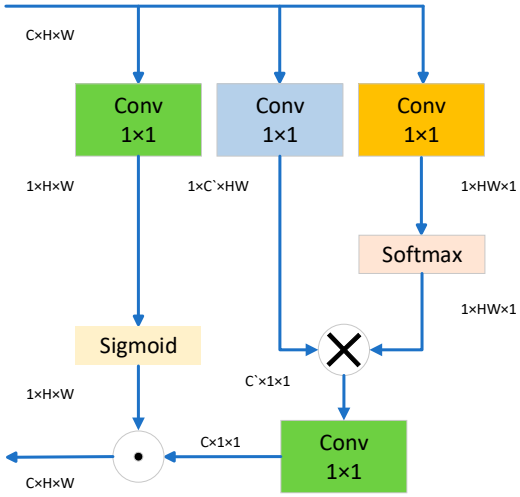


Figure 10. Context aggregation block (Softmax represents the normalization function; the orange convolution is used for generating attention maps; the blue convolution is used for feature mapping; the green convolution is used for contextual refinement; the ⊗ represents batched matrix multiplication; The ⊙ represents broadcast hadamard product; the tensor shape represented by (C, H, W) stands for (channel, height, width)).

3.2.7. CoordConv

To address the issue of morphological differences in apples due to different occlusion types, this research adds a CoordConv layer after each CABlock, as shown in Figure 11. Incorporating pixel position information into the input feature maps enriches spatial feature information in the images, thus improving the model's ability to detect apples under various occlusion scenes. CoordConv [38] enhances the input feature map by adding two coordinate channels, namely, the i-coordinate and j-coordinate channels, to capture spatial information, followed by a standard convolution operation. Regarding whether translation invariance truly contributes to model performance, there is a debate for many tasks [39]. CoordConv allows the model to maintain or discard the translation invariance of traditional convolutions based on learning conditions, reducing the impact caused by the translation invariance of traditional convolutions. By adding CoordConv, we can obtain feature maps with coordinate positional information and increase the number of channels, thereby enhancing feature diversity. As shown in Figure 12, after adding CoordConv, the model significantly enhanced the focus area of the object.

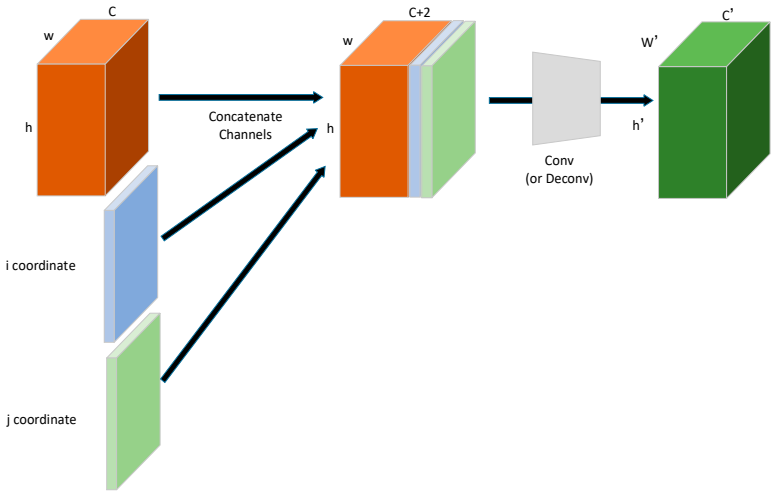
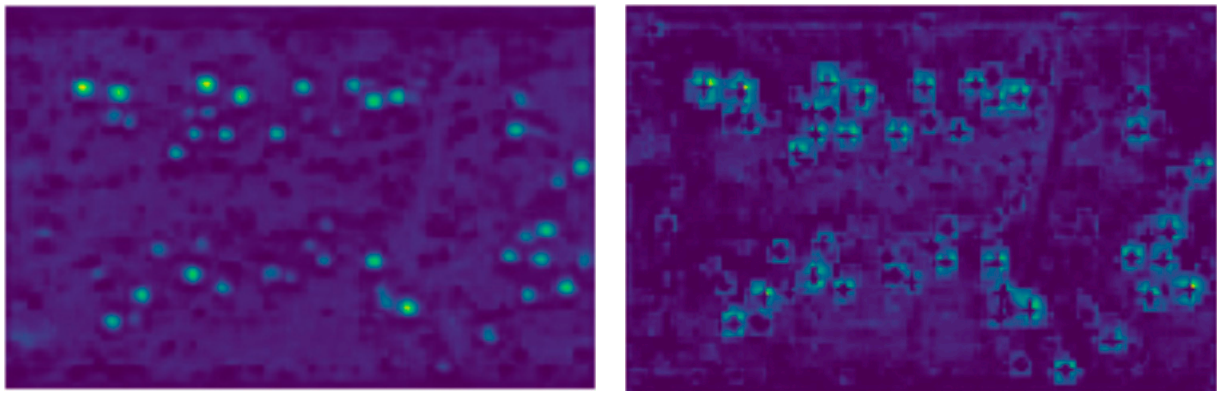


Figure 11. CoordConv. (C, H, W represents the shape of a tensor in terms of channel, height, and width).



(**a**) Stage34_ContextAggregation features (**b**) stage 35_CoordConv features

Figure 12. (**a**) for feature maps without CoordConv; (**b**) for feature maps with CoordConv added.

3.3. Experimental Environment and Parameter Settings

3.3.1. Experimental Platform and Parameter Settings

The improved model and the comparative model in this experiment were trained and validated on an Ubuntu 18.04.3 server. The server was equipped with an Intel(R) Xeon(R) Gold 6140 CPU @ 2.30 GHz processor and an NVIDIA Tesla V100 SXM2 32 GB GPU.

The models involved in this paper were built using PyTorch 1.7.1, a deep learning framework, and CUDA version 10.1. The ablation experiments used the Adam optimization algorithm, with a training duration of 300 epochs. The input images for the models had a resolution of 660 × 640 pixels. The batch size was 32 and the initial learning rate was set to 0.01. A warm-up training of 3 epochs was conducted initially, with the momentum parameter of Adam set to 0.8.

3.3.2. Pre-Trained Model

Pre-trained models refer to a set of network weights that are shared by researchers with others after obtaining good training results from the model. In this paper, YOLOv5-spd-s was used as a pre-trained model for the improved model. The base model used the official pre-trained model provided by YOLOv5. This allowed training to continue in a model with good network weights.

3.3.3. Model Evaluation Indicators

To compare the effect of different models for apple detection in orchards and verify the effectiveness of model improvement, precision, recall, and mean average precision (mAP) were selected for model evaluation: AP is the average precision, which is calculated by combining recall and precision. mAP is used to measure the performance of multiple class label predictions. mAP_0.5 represents the average accuracy of all categories when the IOU value is set to 0.5. mAP_0.5: 0.95 represents the average accuracy of all categories under different IOU values (from 0.5 to 0.95 in steps of 0.05). P represents the precision, i.e., the probability of samples predicted as positive actually being positive. R represents the recall, which is the probability of samples that are actually positive being predicted as positive.

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} = \frac{\text{TP}}{\text{\# predictions}} \tag{6}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} = \frac{\text{TP}}{\text{\# ground truths}} \tag{7}$$

$$\text{mAP\_0.5} = \frac{\sum_{\text{i}}^{\text{K}} \text{AP}_{\text{i}}(\text{IOU} \geq 0.5)}{\text{K}} \tag{8}$$

$$\text{mAP\_0.5 : 0.95} = \frac{\sum_{\text{i}}^{\text{K}} \text{AP}_{\text{i}}(\text{IOU} \geq 0.5 \text{ to } 0.95)}{\text{K}} \tag{9}$$

$$\text{IOU} = \frac{\text{Area of Overlap}}{\text{Area of Union}} \tag{10}$$

$$\sum_{\text{i}}^{\text{K}} \text{AP}_{\text{i}} = \int_0^1 \text{Precision}(\text{Recall}) \text{dRecall} \tag{11}$$

where TP denotes that the input is a positive sample, and the prediction is a positive sample; FN denotes that the input is a positive sample, but the prediction is a negative sample; TN denotes that the input is a negative sample, and the prediction is a negative sample; FP denotes that the input is a negative sample, but the prediction is a positive sample. K represents the number of classes in the dataset, and in this paper, there is only one class, which is 'Apple', so K = 1.

4. Results and Analysis

4.1. Comparison with Other Deep Learning Models

To verify the detection capability of the YOLOv5-ACS model, this paper conducted comparative experiments on the complex scene dataset of annotated apple images. The YOLOv5-ACS model was compared with the mainstream two-stage network Faster RCNN, the one-stage network SSD, and YOLOv7, as shown in Table 4.

Table 4. Results of different object detection models.

Model	Precision (%)	Recall (%)	mAP_0.5 (%)	mAP_0.5:0.95 (%)
Faster RCNN	73.4	92.2	90.6	48.7
SSD	94.3	29.6	76.1	36.6
YOLOv5s	95.0	92.9	97.7	71.6
YOLOv7	95.0	93.0	97.7	70.8
YOLOv5-ACS	95.1	93.9	98.3	74.3

According to Table 4, the YOLOv5-ACS in this experiment showed significant improvements in average precision compared to Faster RCNN, SSD, YOLOv5s, and YOLOv7. The accuracy and recall were also more balanced. Compared to Faster RCNN, SSD, YOLOv5s, and YOLOv7, the accuracy increased by 21.7%, 0.8%, 0.1%, and 0.1%, respectively. The

recall increased by 1.7%, 64.3%, 1%, and 0.9%, respectively. The mAP_0.5 increased by 7.7%, 22.2%, 0.6%, and 0.6%, respectively. The mAP_0.5:0.95 increased by 25.6%, 37.7%, 2.7%, and 3.5%, respectively.

4.2. Ablation Experiments

Ablation experiments on the built complex scene dataset were used to validate the efficacy of the enhanced modules in the YOLOv5s model. Table 5 displays the results of the experiment. In the table, SPD stands for the spatial down-sampling module, C3SE represents the module that introduces channel attention (SENet) in C3, CAM denotes the context augmentation module, CABlock represents the context aggregation module, and CoordConv refers to the coordinate convolution module.

Table 5. Comparison of ablation experiment performance.

	Model	Precision (%)	Recall (%)	mAP_0.5 (%)	mAP_0.5:0.95 (%)
Backbone	YOLOv5s	95.0	92.9	97.7	71.6
	YOLOv5s + SPD	94.7	94.0	97.9	72.2
	YOLOv5s + SPD + C3SE	94.7	94.4	98.0	73.1
	YOLOv5s + SPD + C3SE + CAM	95.3	94.0	98.0	73.2
Neck	YOLOv5s + P2	95.0	92.8	97.9	72.7
Neck + Head	YOLOv5s + P2 + CABlock	94.2	93.6	98.0	72.9
	YOLOv5s + P2 + CABlock + CoordConv	95.0	93.0	98.0	72.7
Backbone + Neck + Head	YOLOv5s + SPD + C3SE + CAM + P2 + CABlock	94.7	94.0	98.2	74.1
	YOLOv5s + SPD + C3SE + CAM + P2 + CABlock + Conv	95.1	93.8	98.2	74.1
	YOLOv5s + SPD + C3SE + CAM + P2 + CABlock + CoordConv	95.1	93.9	98.3	74.3

Table 5 illustrates that the modifications in this paper resulted in notable improvements. Adding SPD-Conv to the YOLOv5s backbone network led to a 0.6% increase in mAP. Further incorporating SPD-Conv and C3SE resulted in a 1.5% mAP improvement. Adding SPD-Conv, C3SE, and CAM further boosted the mAP by 1.6%. By incorporating P2 shallow features into the multi-scale feature fusion of the neck, and adding CABlock and CoordConv in the head, the mAP improved by 1.1%. After incorporating all the improvement modules, the mAP increased by 2.7%, resulting in a significant improvement in average precision.

4.3. Multi-Scale Object Comparison Experiments

As shown in Table 6, by adding SPD-Conv and C3SE, the average precision (AP) and average recall (AR) of small objects were improved by 2.6% and 5.2%, respectively, indicating a significant enhancement in detecting small objects and validating the effectiveness of SPD-Conv and C3SE for small object detection. The YOLOv5-ACS model showed a notable increase in AP and AR for small objects by 10% and 12.5%, respectively, demonstrating the significant impact of all the improved modules in enhancing the perception of small objects.

Table 6. Comparison of multi-scale object detection results.

Model	APs (%)	APm (%)	APl (%)	ARs (%)	ARm (%)	ARl (%)
YOLOv5s	19.0	67.0	83.0	27.1	70.8	86.0
YOLO5s + SPD + C3SE	21.6	68.3	83.9	32.3	72.0	86.8
YOLOv5-ACS	29.0	69.7	85.0	39.6	73.6	87.7

4.4. Detection of Fruits on the Apple Tree in Complex Scenes

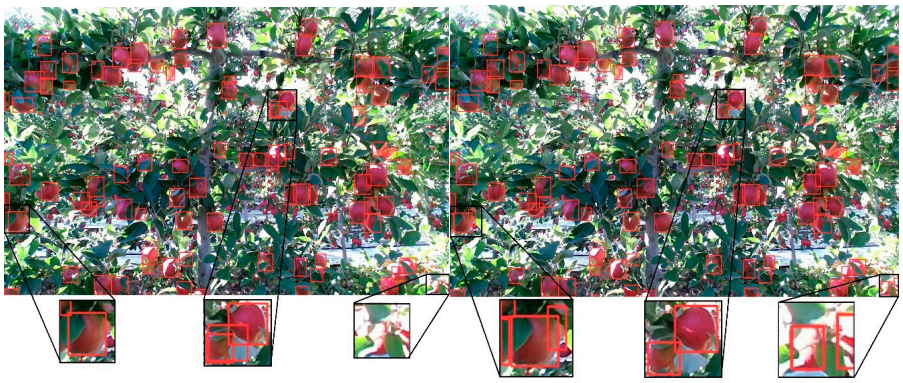
In modern orchards, the complex background, varying apple sizes and ripenesses, the dense arrangement, obstructions from branches and leaves, lighting variations, different image resolutions, and motion blur all pose challenges to apple detection.

Therefore, to evaluate the detection capability of the YOLOv5-ACS model in complex scenes of actual orchards, the trained YOLOv5-ACS model was tested separately on datasets in different scenes. Table 7 displays the test results.

Table 7. Detection results for Different Scenes.

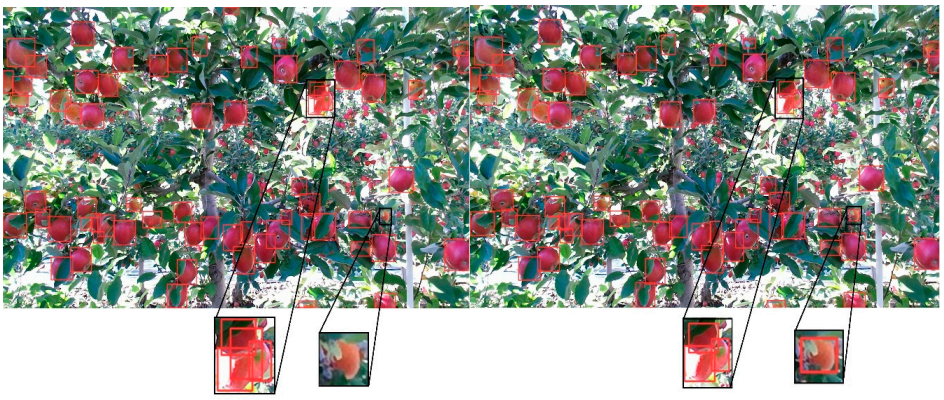
Model	Scenes	Precision (%)	Recall (%)	mAP_0.5 (%)	mAP_0.5:0.95 (%)
YOLOv5s	Nighttime	91.6	92.3	96.6	75.1
YOLOv5-ACS		91.7	93.3	97.2	78.1
YOLOv5s	Nighttime motion blur	89.9	89.2	94.7	68.2
YOLOv5-ACS		90.4	89.6	95.2	70.9
YOLOv5s	Daytime	95.7	94.8	98.4	74.8
YOLOv5-ACS		95.8	95.8	98.9	77.6
YOLOv5s	Daytime motion blur	94.8	91.6	97.1	69.4
YOLOv5-ACS		93.7	93.5	97.8	72.2
YOLOv5s	Dimming	95.7	93.7	98.1	73.4
YOLOv5-ACS		95.7	95.0	98.7	76.2
YOLOv5s	Brightening	95.8	94.5	98.3	74.7
YOLOv5-ACS		96.0	95.6	98.8	77.4

By comparing YOLOv5-ACS with the original model (Figures 13–18), the YOLOv5-ACS model can be seen to produce better overall detection results, especially in terms of small objects, severe occlusion, strong lighting conditions, dim lighting, and image edges. The YOLOv5-ACS model demonstrates better performance in reducing false positives and false negatives. It was proven that the YOLOv5-ACS network has a better generalization performance in complex orchard scenes.



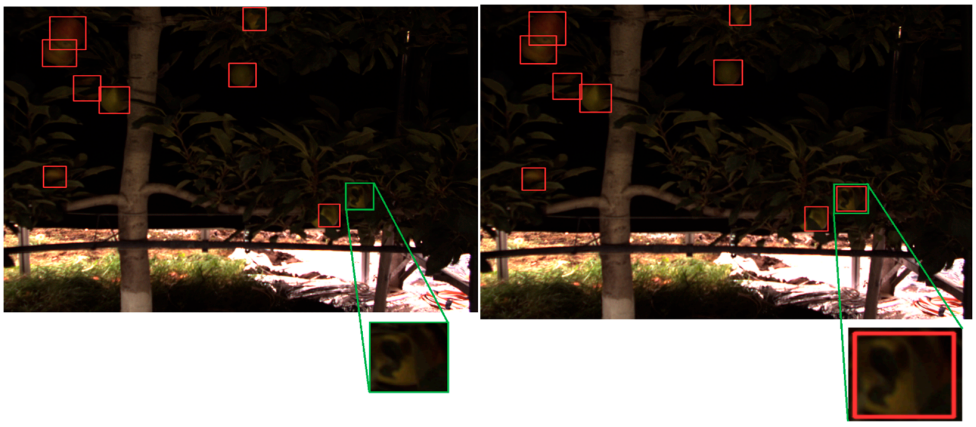
(**a**) (**b**)

Figure 13. Compared to the YOLOv5-ACS (**b**), the YOLOv5 (**a**) exhibited the following shortcomings: (1) it missed two apple targets in each of the cases of branch or leaf occlusion and overexposure, and (2) it generated redundant bounding boxes when tree leaves were obstructing the view.



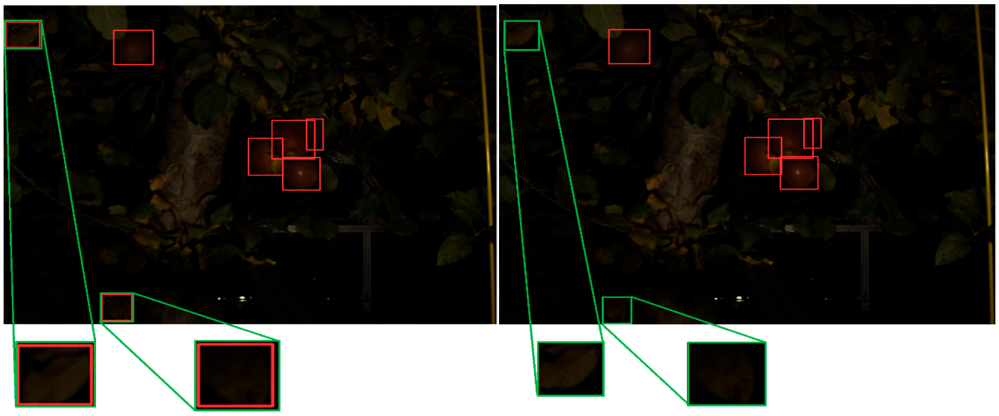
(**a**) (**b**)

Figure 14. Compared to the YOLOv5-ACS (**b**), the YOLOv5 (**a**) exhibited the following shortcomings: (1) it produced redundant bounding boxes when fruits overlapped, and (2) it missed one small apple target during detection.



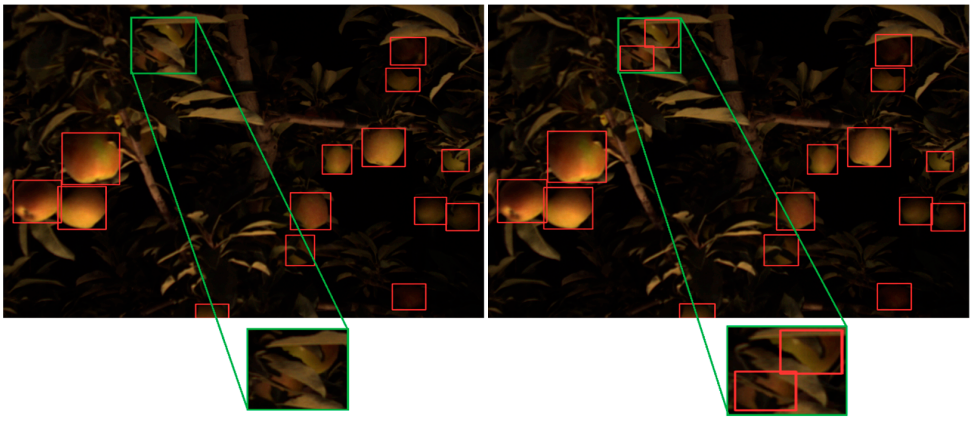
(**a**) (**b**)

Figure 15. Compared to the YOLOv5-ACS (**b**), the YOLOv5 (**a**) exhibited the following shortcomings: it missed one target during the nighttime detection of green apples.



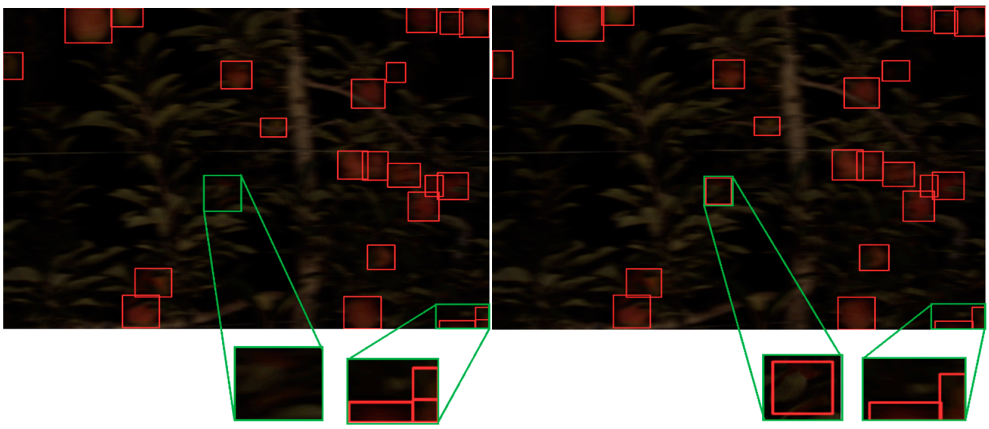
(**a**) (**b**)

Figure 16. Compared to the YOLOv5-ACS (**b**), the YOLOv5 (**a**) exhibited the following shortcomings: it mistakenly predicted leaves as apple targets in darker nighttime conditions.



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

Figure 17. Compared to the YOLOv5-ACS (**b**), the YOLOv5 (**a**) exhibited the following shortcomings: it missed two apple targets in nighttime artificial lighting scenes with leaf occlusion.



(**a**) YOLOv5s (**b**) YOLOv5-ACS

Figure 18. Compared to the YOLOv5-ACS (**b**), the YOLOv5 (**a**) exhibited the following shortcomings: (1) it missed one apple target in nighttime motion blur scenes with leaf occlusion, and (2) it generated redundant bounding boxes in nighttime motion blur scenes.

4.5. Holdout Cross-Validation

To confirm the model's dependability and capacity for generalization, we randomly split the dataset using different random seeds. The ratio of the training set, validation set, and test set was 6:2:2. As displayed in Table 8, the performance results of the model are almost consistent, demonstrating its reliability.

Table 8. Results under different dataset partitioning.

Model	Random Seed	Precision (%)	Recall (%)	mAP_0.5 (%)	mAP_0.5:0.95 (%)
YOLOv5s	1	95.2	92.8	97.4	71.7
YOLOv5s-ACS		95.4	93.7	98.1	74.6
YOLOv5s	2	94.8	93.0	97.0	71.6
YOLOv5s-ACS		95.0	93.6	98.1	74.5
YOLOv5s	3	95.4	92.4	97.3	71.4
YOLOv5s-ACS		95.7	93.1	98.1	74.4

5. Conclusions

In response to the challenges of complex backgrounds, numerous small objects, and severe occlusion in apple images in modern orchard environments, this study integrates multi-scene data and proposes a YOLOv5-ACS algorithm. Based on the YOLOv5s model, the algorithm incorporates the SPD-Conv module and C3SE module to enhance the perception of small objects. The context augmentation module is used to aggregate information from different receptive fields to provide rich contextual information for the FPN. The P2 feature layer is fused to retain the low-level features of nighttime apple images. The context aggregation module and CoordConv are employed to aggregate pixel-level contextual information and enhance the detection capabilities of small objects by further improving feature channels.

This study examines the detection performance of the YOLOv5-ACS model and the original model across various scales, demonstrating that the YOLOv5-ACS model remarkably improves the detection accuracy for small objects. The detection performance of the YOLOv5-ACS model is also compared across different datasets, confirming its good generalization ability. Various experiments verified that the YOLOv5 ACS model is superior to other models, with a precision of 95.108, a recall of 93.753, a mAP_0.5 of 98.223, and a mAP_0.5:0.95 of 74.265.

This study proposes a new model for detecting apples on trees in complex scenes. Although our model performed well in the various complex scenarios mentioned in the paper, it lacks deployment capabilities for mobile applications. Future research can con-

sider integrating this model as the visual component of automated hardware, deploying it to mobile devices, and providing model and algorithm support for practical industry issues such as automated harvesting, automated spraying, yield estimation, and growth monitoring.

Author Contributions: Conceptualization, J.L. and C.W.; methodology, J.L. and C.W.; software, J.X.; validation, C.W. and J.X.; formal analysis, J.L. and C.W.; investigation, J.L. and C.W.; data curation, C.W.; writing—original draft preparation, J.L. and C.W.; writing—review and editing, C.W.; visualization, J.X.; funding acquisition, J.L. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported in part by the Key Research and Development Program for Talent Introduction of Ningxia Province China titled "Research on Key Technologies of Scientific Data Retrieval in the Context of Open Science" under Grant 2022BSB03044; in part by the Natural Science Foundation Project of Ningxia Province, China, titled "User-Oriented Multi-Criteria Relevance Ranking Algorithm and Its Application" under Grant 2021AAC03205; in part by the Starting Project of Scientific Research in the North Minzu University titled "Research of Information Retrieval Model Based on the Decision Process" under Grant 2020KYQD37.

Data Availability Statement: The daytime dataset used in this paper can be downloaded from GitHub (https://github.com/fu3lab/Scifresh-apple-RGB-images-with-multi-class-label, accessed on 29 August 2020), while the nighttime dataset is available for download on WSU Research Exchange [40].

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

References

1. United States Department of Agriculture. *Fresh Apples, Grapes, and Pears: World Markets and Trade*; Foreign Agricultural Service: Washington, DC, USA, 2019; pp. 1–10.
2. Yue, Y.; Tian, K.; Wang, H.; Zhao, H. Research on apple detection in complex environment based on improved Mask RCNN. *J. Chin. Agric. Mech.* **2019**, *40*, 128–134.
3. Bhagya, C.; Shyna, A. An Overview of Deep Learning Based Object Detection Techniques. In Proceedings of the 2019 1st International Conference on Innovations in Information and Communication Technology (ICIICT), Chennai, India, 25–26 April 2019; pp. 1–6.
4. Chu, P.; Li, Z.; Zhang, K.; Chen, D.; Lammers, K.; Lu, R. O2RNet: Occluder-Occludee Relational Network for Robust Apple Detection in Clustered Orchard Environments. *arXiv* **2023**, arXiv:2303.04884. [CrossRef]
5. Sun, M.; Xu, L.; Chen, X.; Ji, Z.; Zheng, Y.; Jia, W. BFP Net: Balanced Feature Pyramid Network for Small Apple Detection in Complex Orchard Environment. *Plant Phenomics* **2022**, *2022*, 9892464. [CrossRef] [PubMed]
6. Xuan, G.; Gao, C.; Shao, Y.; Zhang, M.; Wang, Y.; Zhong, J.; Li, Q.; Peng, H. Apple Detection in Natural Environment Using Deep Learning Algorithms. *IEEE Access* **2020**, *8*, 216772–216780. [CrossRef]
7. Meng, H. *Optimized Detection Algorithm for Green Fruit Based on Attention Mechanism*; Shandong Normal University: Jinan, China, 2023.
8. Shf, P.; Zhao, C. Review on Deep Based Object Detection. In Proceedings of the 2020 International Conference on Intelligent Computing and Human-Computer Interaction (ICHCI), Sanya, China, 4–6 December 2020; pp. 372–377.
9. Peng, X.; Yu, X.; Luo, Y.; Chang, Y.; Lu, C.; Chen, X. Prediction Model of Greenhouse Tomato Yield Using Data Based on Different Soil Fertility Conditions. *Agronomy* **2023**, *13*, 1892. [CrossRef]
10. Girshick, R.; Donahue, J.; Darrell, T.; Malik, J. Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014; pp. 580–587.
11. Ren, S. Fast r-cnn. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 1440–1448.
12. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **2016**, *39*, 1137–1149. [CrossRef]
13. He, K.; Zhang, X.; Ren, S.; Sun, J. Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2015**, *37*, 1904–1916. [CrossRef]
14. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You Only Look Once: Unified, Real-Time Object Detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 779–788.
15. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.-Y.; Berg, A.C. SSD: Single Shot MultiBox Detector. In *Computer Vision–ECCV 2016, Proceedings of the 14th European Conference, Amsterdam, The Netherlands, 11–14 October 2016*; Springer: Cham, Switzerland, 2016; pp. 21–37.

16. Lin, T.-Y.; Goyal, P.; Girshick, R.; He, K.; Dollár, P. Focal Loss for Dense Object Detection. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 2980–2988.
17. Carion, N.; Massa, F.; Synnaeve, G.; Usunier, N.; Kirillov, A.; Zagoruyko, S. End-to-End Object Detection with Transformers. In Proceedings of the European Conference on Computer Vision, Glasgow, UK, 23–28 August 2020; pp. 213–229.
18. Zhu, X.; Su, W.; Lu, L.; Li, B.; Wang, X.; Dai, J. Deformable DETR: Deformable Transformers for End-to-End Object Detection. *arXiv* **2021**, arXiv:2010.04159.
19. Yuan, X.; Ma, X.; Liu, S. An Improved Algorithm of Pedestrian and Vehicle Detection Based on YOLOv3. *Sci. Technol. Eng.* **2021**, *21*, 3192–3198.
20. Song, X.; Zhang, D.; Zhang, P.; Liang, L.; Hei, X. Real-time object detection algorithm for complex construction environments. *J. Comput. Appl.* **2023**, 1–9. [CrossRef]
21. Li, Q.; Yang, X.; Lu, R.; Wang, S.; Xie, X.; Zhang, T. Transformer in Computer Vision: A Survey. *J. Chin. Mini-Micro Comput. Syst.* **2023**, *44*, 850–861.
22. Li, Y.; Xue, J.; Zhang, M.; Yin, J.; Liu, Y.; Qiao, X.; Zheng, D.; Li, Z. YOLOv5-ASFF: A Multistage Strawberry Detection Algorithm Based on Improved YOLOv5. *Agronomy* **2023**, *13*, 1901. [CrossRef]
23. Li, Y.; Li, X.; Hu, Z.; Su, X.; Chen, F. The research on lightweight SAR ship detection method based on regression model and attention. *J. Infrared Millim. Waves* **2022**, *41*, 618–625.
24. Dong, W.; Liang, H.; Liu, G.; Hu, Q.; Yu, X. Review of Deep Convolution Applied to Target Detection Algorithms. *J. Front. Comput. Sci. Technol.* **2022**, *5*, 1025–1042.
25. Peng, C.; Zhang, Q.; Tang, Z.; Gui, W. Research on Mask Wearing Detection Method Based on YOLOv5 Enhancement Model. *Comput. Eng.* **2022**, *48*, 39–49.
26. Hu, D.; Zhang, Z. Road target detection algorithm for autonomous driving scenarios based on improved YOLOv5s. *CAAI Trans. Intell. Syst.* **2023**, 1–9. Available online: http://kns.cnki.net/kcms/detail/23.1538.TP.20230913.1825.004.html (accessed on 8 October 2023).
27. Zhou, H.; Ou, J.; Meng, P.; Tong, J.; Ye, H.; Li, Z. Reasearch on Kiwi Fruit Flower Recognition for Efficient Pollination Based on an Improved YOLOv5 Algorithm. *Horticulturae* **2023**, *9*, 400. [CrossRef]
28. Mu, L.; Gao, Z.; Cui, Y.; Li, K.; Liu, H.; Fu, L. Kiwifruit Detection of Far-view and Occluded Fruit Based on Improved AlexNet. *Trans. Chin. Soc. Agric. Mach.* **2019**, *50*, 24–34.
29. Li, C.; Wang, S. Identification and Detection of Picking Targets of Orah Mandarin Orange in Natural Environment Based on SSD Model. In Proceedings of the 2021 IEEE 3rd Eurasia Conference on IOT, Communication and Engineering (ECICE), Yunlin, Taiwan, 29–31 October 2021; pp. 439–442.
30. Wu, X.; Qi, Z.; Wang, L.; Yang, J.; Xia, X. Apple Detection Method Based on Light-YOLOv3 Convolutional Neural Network. *Trans. Chin. Soc. Agric. Mach.* **2020**, *51*, 17–25.
31. Li, Y.; Rao, Y.; Jin, X.; Jiang, Z.; Wang, Y.; Wang, T.; Wang, F.; Luo, Q.; Liu, L. YOLOv5s-FP: A Novel Method for In-Field Pear Detection Using a Transformer Encoder and Multi-Scale Collaboration Perception. *Sensors* **2023**, *23*, 30. [CrossRef]
32. Chen, J.; Ma, A.; Huang, L.; Su, Y.; Li, W.; Zhang, H.; Wang, Z. GA-YOLO: A Lightweight YOLO Model for Dense and Occluded Grape Target Detection. *Horticulturae* **2023**, *9*, 443. [CrossRef]
33. Qiu, Z.; Zeng, J.; Tang, W.; Yang, H.; Lu, J.; Zhao, Z. Research on Real-Time Automatic Picking of Ground-Penetrating Radar Image Features by Using Machine Learning. *Horticulturae* **2022**, *8*, 1116. [CrossRef]
34. Sunkara, R.; Luo, T. No More Strided Convolutions or Pooling: A New CNN Building Block for Low-Resolution Images and Small Objects. *arXiv* **2022**, arXiv:2208.03641.
35. Hu, J.; Shen, L.; Sun, G. Squeeze-and-Excitation Networks. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 7132–7141.
36. Xiao, J.; Zhao, T.; Yao, Y.; Yu, Q.; Chen, Y. Context Augmentation and Feature Refinement Network for Tiny Object Detection. 2021. Available online: https://openreview.net/forum?id=q2ZaVU6bEsT (accessed on 6 September 2023).
37. Liu, Y.; Li, H.; Hu, C.; Luo, S.; Luo, Y.; Chen, C.W. Learning to Aggregate Multi-Scale Context for Instance Segmentation in Remote Sensing Images. *arXiv* **2021**, arXiv:2111.11057.
38. Liu, R.; Lehman, J.; Molino, P.; Such, F.P.; Frank, E.; Sergeev, A.; Yosinski, J. An Intriguing Failing of Convolutional Neural Networks and the CoordConv Solution. In Proceedings of the 2018 Conference on Neural Information Processing Systems, Montreal, QC, Canada, 3–8 December 2018.
39. Du, J.; Cui, S.; Jin, M.; Ru, C. Improved the Complex Road Scene Object Detection Algorithm of YOLOv7. *Comput. Eng. Appl.* **2023**, 1–12. Available online: http://kns.cnki.net/kcms/detail/11.2127.TP.20230811.1710.026.html (accessed on 26 August 2023).
40. Bhusal, S.; Karkee, M.; Zhang, Q. *Apple Dataset Benchmark from Orchard Environment in Modern Fruiting Wall*; Washington State University: Pullman, WA, USA, 2019. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.