agronomy

MDPI

Article

New Progress in Intelligent Picking: Online Detection of Apple Maturity and Fruit Diameter Based on Machine Vision

Junsheng Liu [1], Guangze Zhao [1], Shuangxi Liu [1,2], Yi Liu [1], Huawei Yang [1,3], Jingwei Sun [1], Yinfa Yan [1,4], Guoqiang Fan [1,2], Jinxing Wang [1,4] and Hongjian Zhang [1,2,4,*]

[1] College of Mechanical and Electrical Engineering, Shandong Agricultural University, Taian 271002, China; 2021110421@sdau.edu.cn (J.L.); 17686277809@163.com (G.Z.); shuangxiliu168@163.com (S.L.); 17863801358@163.com (Y.L.); 2022120708@sdau.edu.cn (H.Y.); sunjw0714@163.com (J.S.); guanghaok@gmail.com (Y.Y.); fgq1217@163.com (G.F.); jinxingw@163.com (J.W.)
[2] Shandong Provincial Key Laboratory of Horticultural Machinery and Equipment, Taian 271018, China
[3] Shandong Academy of Agricultural Machinery Sciences, Jinan 250010, China
[4] Shandong Agricultural Equipment Intelligent Engineering Laboratory, Taian 271002, China
* Correspondence: zhanghongji_an@163.com

Abstract: In the realm of automated apple picking operations, the real-time monitoring of apple maturity and diameter characteristics is of paramount importance. Given the constraints associated with feature detection of apples in automated harvesting, this study proposes a machine vision-based methodology for the accurate identification of Fuji apples' maturity and diameter. Firstly, maturity level detection employed an improved YOLOv5s object detection model. The feature fusion section of the YOLOv5s network was optimized by introducing the cross-level partial network module VoVGSCSP and lightweight convolution GSConv. This optimization aimed to improve the model's multiscale feature information fusion ability while accelerating inference speed and reducing parameter count. Within the enhanced feature fusion network, a dual attention mechanism combining channel and spatial attention (GAM) was introduced to refine the color and texture feature information of apples and to increase spatial position feature weights. In terms of diameter determination, the contours of apples are obtained by integrating the dual features of color and depth images within the target boxes acquired using the maturity detection model. Subsequently, the actual area of the apple contour is determined by calculating the conversion relationship between pixel area and real area at the current depth value, thereby obtaining the diameter of the apples. Experimental results showed that the improved YOLOv5s model achieved an average maturity level detection precision of 98.7%. Particularly noteworthy was the detection accuracy for low maturity apples, reaching 97.4%, surpassing Faster R-CNN, Mask R-CNN, YOLOv7, and YOLOv5s models by 6.6%, 5.5%, 10.1%, and 11.0% with a real-time detection frame rate of 155 FPS. Diameter detection achieved a success rate of 93.3% with a real-time detection frame rate of 56 FPS and an average diameter deviation of 0.878 mm for 10 apple targets across three trials. Finally, the proposed method achieved an average precision of 98.7% for online detection of apple maturity level and 93.3% for fruit diameter features. The overall real-time inference speed was approximately 56 frames per second. These findings indicated that the method met the requirements of real-time mechanical harvesting operations, offering practical importance for the advancement of the apple industry.

Keywords: apple; computer vision; YOLOv5; ripeness; fruit diameter; online detection

check for updates

Citation: Liu, J.; Zhao, G.; Liu, S.; Liu, Y.; Yang, H.; Sun, J.; Yan, Y.; Fan, G.; Wang, J.; Zhang, H. New Progress in Intelligent Picking: Online Detection of Apple Maturity and Fruit Diameter Based on Machine Vision. *Agronomy* **2024**, *14*, 721. https://doi.org/10.3390/agronomy14040721

Academic Editor: Yanbo Huang

Received: 3 March 2024
Revised: 21 March 2024
Accepted: 29 March 2024
Published: 31 March 2024

Copyright: © 2024 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

As one of the four largest fruits in the world, apples are beloved by global consumers for their rich vitamins and minerals. In the apple harvesting process, online detection of ripeness and diameter features plays a critical role, directly influencing decisions related to harvesting, packaging, transportation, storage methods, and market pricing [1,2]. For orchard owners, real-time detection of apple ripeness and diameter enables the prediction of

different-grade fruit yields, leading to labor efficiency and enhanced economic benefits [3]. For distributors, accurate screening of defective apples aids in mitigating the incidence of decay and deterioration resulting from inadequate storage, thereby reducing financial losses [4]. Grading apples according to varying levels of ripeness contributes to the optimization of storage strategies and increases in market share [5], while grading based on different diameters aids in optimizing packaging, transportation, and meeting diverse market demands [6]. Consumers typically assess the quality of fruit based on external attributes such as color, size, shape, and surface imperfections. Therefore, apple grading contributes to enhancing product appeal [7,8]. Online detection of apple ripeness and diameter features is essential for the realization of automated apple harvesting, thereby enhancing market competitiveness and maximizing the economic benefits of the apple industry.

Traditional machine learning methods have shown effectiveness in the quality grading of agricultural products such as apples [9], pears [10], and citrus fruits [11]. Moallem et al. [12] proposed a computer vision-based grading algorithm designed for Golden Delicious apples. After removing stem and calyx regions from the defect area, statistical, textural, and geometric features were extracted. Then, a support vector machine (SVM) algorithm was used for classification, achieving average recognition accuracies of 92.5% for two categories (healthy and defective) and 89.2% for three quality categories (first grade, second grade, and reject). Hu et al. [13] developed an on-site apple detection and grading device integrating four features—size, color, shape, and surface defects. Then, the four apple features were fused, and a support vector machine was used for infield apple grading into three grades: first-grade fruit, second-grade fruit, and other-grade fruit. The results showed that, for a single index, the accuracies of detecting the apple size, the fruit shape, the color, and the surface defects were 99.04%, 97.71%, 98%, and 95.85%. The grading accuracies for the first-grade fruit, second-grade fruit, other-grade fruit, and the average grading accuracy based on multiple features were 94.55%, 95.71%, 100%, and 95.49%, respectively. While the aforementioned apple grading methods based on traditional machine learning achieved satisfactory results under specific conditions, they face challenges in recognizing apples in the complex real-world environment of orchards. These challenges include different lighting conditions, various ripeness levels, multiple fruit types, and backgrounds of varying complexity. As a result, significant environmental interference compromises both detection accuracy and speed, rendering these methods inadequate to meet the demands of automated apple harvesting.

With the continuous advancement of deep learning theories and the rapid development of hardware devices, end-to-end neural network processing methods are increasingly applied in the field of agricultural automation. This provides new insights into online feature detection for apples [14,15]. In order to address the challenges posed by complex orchard environments on the accuracy of object detection and achieve high-precision detection of fruit ripeness and diameter, two-stage object detectors are commonly utilized in the field of online fruit grading. Wang et al. [16] developed a method for precise segmentation of apple instances based on improved Mask RCNN, categorizing apples into immature, semi-mature, and mature classes and achieving a mAP of 0.917%. For accurate automatic sorting of apple ripeness levels, Zhang et al. [17] proposed a fine-grained lightweight architecture for the Fuji apple ripeness classification (FGAL-MC) based on CNN, enabling precise division of apple ripeness levels in unstructured orchard environments. However, while two-stage object detectors are capable of providing pixel-level high-precision fruit grading predictions, they often do not meet the real-time requirements for online feature detection in apple harvesting robots. Representing a single-stage detector, YOLO has shown significant potential in online fruit detection, attributed to its exceptionally rapid target detection speed [18]. Lou et al. [19] proposed a fruit quality inspection and classification method based on YOLOv5 in which images of four types of fruits, namely apples, oranges, bananas, and pears, were collected to construct a fruit image dataset, and a model was trained subsequently. The model demonstrated an average accuracy of 95.3% in fruit quality detection and classification with an algorithmic inference time of 10.5 milliseconds per image. Li et al. [20] proposed a fruit volume measurement method that

integrates structure from motion (SFM) and deep learning. This method utilizes multi-view images captured by a monocular camera combining structured light field (SFM) technology and neural networks. It facilitates the rapid inference of fruit structure and size estimation, addressing the issue of time-consuming dense point cloud construction in traditional 3D reconstruction. By reconstructing the fruit model under multiple views and conducting actual volume measurement, the method was evaluated. The results indicate that the relative error is generally within 12% with an average error of approximately 7.75%, demonstrating that the algorithm achieves relatively precise fruit volume measurement. Nonetheless, challenges persist in employing deep learning techniques for online fruit grading. Firstly, existing methods for diameter calculation are influenced by the diversity of fruit shapes and contours, making it difficult to accurately capture the diameter of complexly shaped fruits. This may result in some bias in diameter calculation, affecting the accurate discrimination of fruit grades. Secondly, achieving high-precision ripeness discrimination is particularly challenging due to the similarity in morphological contours of fruits at different ripeness levels but varying colors. This is especially true for fruits with low ripeness, where the color resembles that of background leaves under natural conditions, making discrimination more complex [21].

Therefore, addressing the prevalent issue in apple harvesting robots where undifferentiated harvesting of all apples occurs without the ability to distinguish ripeness and diameter results in inefficient harvesting and inconsistent fruit quality. The aim of this study is to achieve maturity level discrimination and fruit diameter measurement of Fuji apples, enhancing the visual capabilities of apple-picking robots. Firstly, we improved the YOLOv5s network model by focusing on optimizing feature extraction and fusion methods at the neck of the network, comprehensively considering various features such as color, texture, and shape of the fruits to capture maturity information comprehensively. This improvement not only maintains the real-time performance of the model but also enhances the accurate discrimination of various maturity levels and defects. Additionally, we introduced a depth-image-based method for fruit diameter calculation to extract more precise apple contours and effectively improve the accuracy of diameter calculation by integrating depth information. These comprehensive improvements enable our model to better adapt to complex orchard environments, ensuring the accuracy and consistency of maturity level and fruit diameter assessments.

2. Datasets and Methods

2.1. Image Collection and Processing

The experimental data used for research were collected from the Fuji apple experimental orchard of Shandong Agricultural University in Tai'an City, Shandong Province, China. To ensure various apple samples at different ripening stages, the sample collection was conducted multiple times between September and October 2022. As shown in Figure 1, color images were captured using a RealSense depth camera, and random sampling was performed at different locations within the orchard to cover various fruit states and environmental conditions. Considering that the working range of the apple picking robot was from −100 mm to 700 mm, and the distance between the camera installation position and the mechanical arm origin was 582 mm, the distance between the camera and the fruit tree was limited to the range of 0.5 m to 1.5 m to ensure that the apples in the images were within the appropriate shooting range. A total of 2000 apple images were collected and saved in .jpg format with a resolution of 1280 × 720 pixels.

To increase sample diversity and improve the robustness and generalization ability of the training model, four data augmentation methods were employed: blurring, darkening, adding noise, and rotation. Two augmentation methods were randomly applied to each image, resulting in a total of 6000 augmented images. The dataset was divided into training, validation, and testing sets in a ratio of 7:2:1. The annotated dataset includes individual samples categorized by high, medium, and low ripeness as well as surface defects, consisting of 3146, 2587, 3419, and 2786 samples, respectively. For detailed information, please refer to Figure 1.

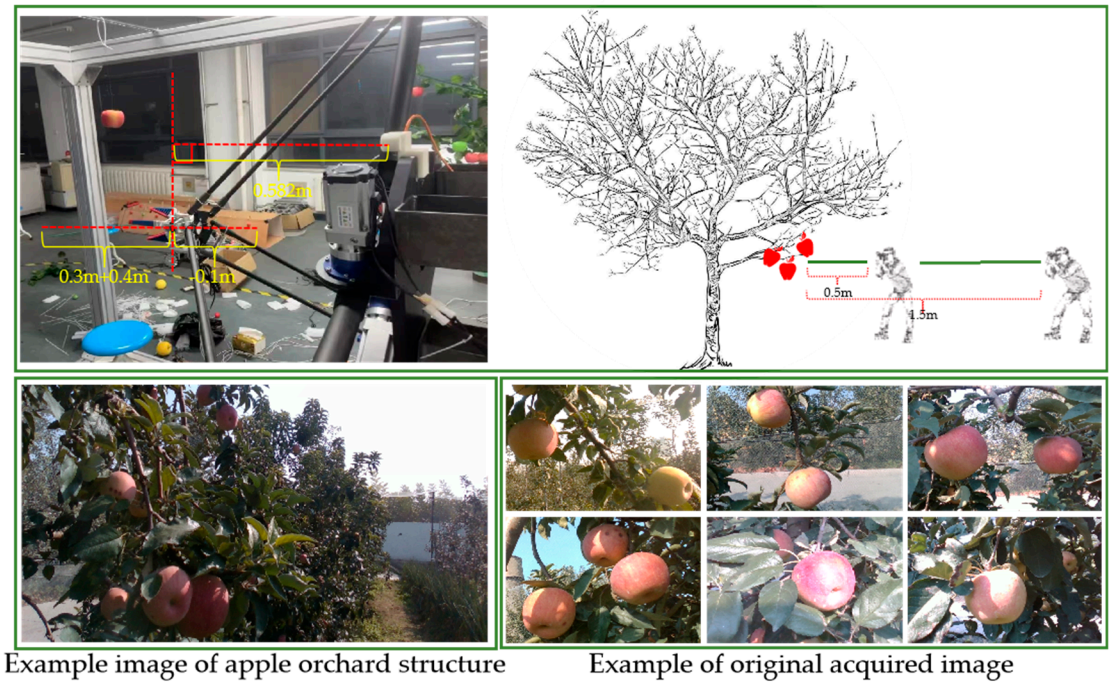


Figure 1. Apple image acquisition process.

2.2. Dataset Division Criteria

In this study, the classification criteria for apple fruit were based on the national standard GB/T 10651-2008 for fresh apples. As illustrated in Table 1, ripeness and fruit diameter were selected as the grading criteria for apples. The LabImg annotation tool was utilized to categorize the apple ripeness dataset, as shown in Figure 2, where label 1 indicates high maturity, label 2 denotes medium maturity, label 3 signifies low maturity, and label 4 represents surface defects. This adheres to the grading standards outlined in the national guidelines.



Figure 2. Division of the apple maturity dataset.

Table 1. Dataset division criteria.

Grades	Criteria
High maturity	Red or striped red covers more than 90%
Medium maturity	Red or striped red covers 80% or more
Low maturity	Red or striped red covers less than 80%
Surface defect	Fruit skin exhibits various damages
Grade-1	Diameter above 85 mm
Grade-2	Diameter from 80 to 85 mm (excluding 85 mm)
Grade-3	Diameter from 75 to 80 mm (excluding 80 mm)
Grade-4	Diameter below 75 mm

2.3. YOLOv5s Network Model Structure

The size of the pre-trained network model plays a crucial role in determining the training accuracy and detection time of the model. Considering the need for real-time precision in apple harvesting and the requirement for a balance between accuracy and speed, the YOLOv5s network, characterized by a smaller architecture and faster detection speed, was selected for apple ripeness detection. The YOLOv5s network utilized in this study is structured with three key components: backbone, neck, and prediction, as illustrated in Figure 3. This choice was made to ensure model accuracy while meeting the demands for real-time apple ripeness detection.

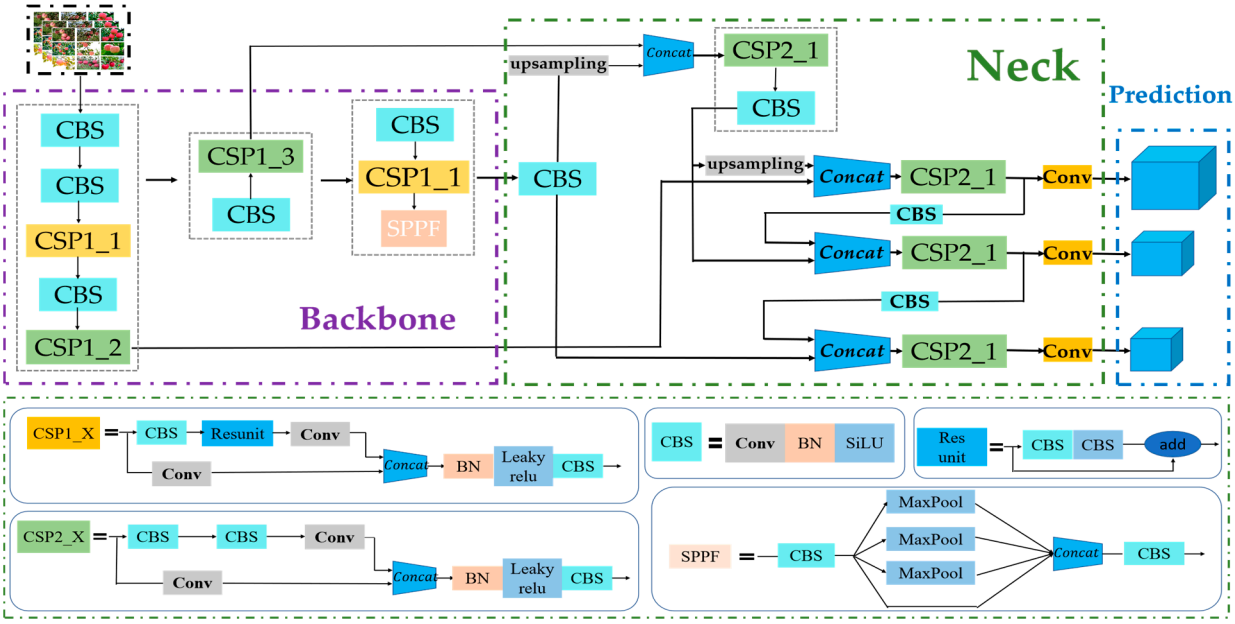


Figure 3. Network architecture of YOLOv5s.

2.4. Improved YOLOv5s Apple Ripeness Detection Model

The original network's neck section suffers from inadequate processing speed and lacks focus on color-specific features, which poses a significant challenge for real-time operation of apple-picking robots, as they require swift and accurate identification of apple ripeness. Therefore, in response to these requirements, we implemented enhancements to the neck section. We replaced the original network's CBS and CSP2_1 modules with GSConv and VoVGSCSP modules to enhance processing speed and to better emphasize color-specific features related to apple ripeness. Additionally, we introduced a global attention mechanism (GAM) to more effectively extract texture information, particularly emphasizing defect categories. The improved model is abbreviated as YOLOv5s-GGV, where the first "G" represents global attention mechanism, the second "G" represents GSConv modules, and "V" represents VoVGSCSP modules. The improved YOLOv5s-GGV

network structure is shown in Figure 4.The motivation behind these improvements is to achieve more precise and faster apple ripeness detection to meet the real-time operational requirements of practical harvesting robots.

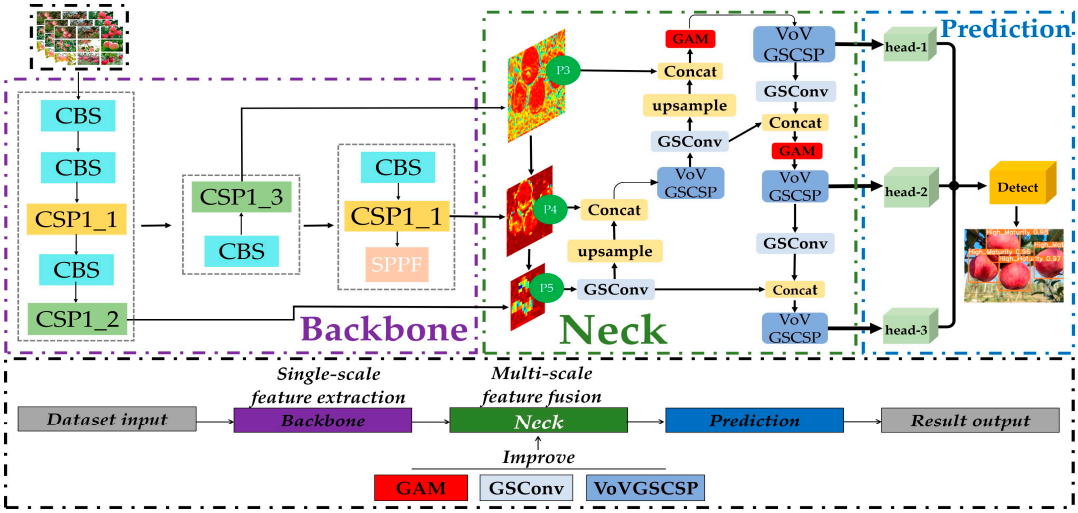


Figure 4. Network architecture of YOLOv5s-GGV (including global attention mechanism, GSConv and VoVGSCSP).

2.4.1. Anchor Box Calculation Using k-Means++

Setting different anchor boxes, which are initial boxes with specific lengths and widths, based on distinct datasets is a crucial step in network training. The YOLOv5s network adaptively computes the optimal anchor box values for various training sets. During network training, the model generates predicted boxes based on initial anchor boxes, compares them with actual boxes, calculates the disparity, and iteratively updates the network parameters in a backward manner.

At the commencement of training, the network calculates the initial anchor box's optimal recall rate (BPR) based on the dataset's label information. The BPR, rated on a scale of 0 to 1, where a higher value signifies superior performance, is automatically updated using the k-means and genetic algorithm in the kmeans_anchors function when BPR is less than 0.98, aligning the anchors more closely with the dataset. However, the k-means algorithm initiates by randomly selecting k points from the dataset as cluster centers, a process that poses challenges for accurate estimation. In contrast, the k-means++ algorithm selects the first cluster center randomly and subsequently chooses the next center with a higher probability of being farther from the previous one. This process repeats until k cluster centers are selected. Therefore, opting for the k-means++ algorithm provides a more reasonable generation of initial anchor boxes.

The anchor boxes for this dataset, calculated using the k-means++ algorithm and rounded to the nearest integers, are presented in Table 2. The three rows correspond to anchors on feature maps with different resolutions, enabling the calculation for large, medium, and small targets. The first column represents data for feature maps with various sampling ratios allocated to images of size 640 × 640 pixels.

Table 2. Anchor box size comparison chart.

Down-Sampling	Anchors Box1 (px)		Anchors Box2 (px)		Anchors Box3 (px)	
	k-Means++	k-Means	k-Means++	k-Means	k-Means++	k-Means
8	24 × 40	10 × 13	35 × 60	16 × 30	47 × 80	23 × 33
16	63 × 108	30 × 61	74 × 130	45 × 62	101 × 171	59 × 119
32	137 × 237	90 × 116	203 × 348	115 × 198	312 × 532	326 × 373

As evident from the table, it is evident that there exists a significant disparity in anchor box dimensions derived through k-means++ in comparison to the original YOLOv5 anchor box dimensions. Specifically, the anchor box dimensions computed by the k-means algorithm for target apples predominantly range from 10 × 13 to 326 × 373 pixels, while those derived through the k-means++ algorithm primarily span from 24 × 40 to 312 × 532 pixels. Considering the actual dataset annotations, it becomes apparent that the anchor box dimensions obtained through the k-means++ algorithm better accommodate the diverse sizes of target apples. Consequently, we opt to employ this methodology for model training.

2.4.2. Improvement of Feature Fusion Network with GSConv and VoVGSCSP

In the feature fusion network of YOLOv5s, Conv, Bottleneck, and C3 modules are employed to process feature information. While these modules enhance the model's detection accuracy and robustness to some extent, they significantly increase the model's inference time and parameter count when dealing with cross-stage partial connections and processing large-sized images. Additionally, when handling larger targets, the C3 module and Conv operation might compromise target details, affecting the model's accuracy and robustness [22].

To address this, this study introduces lightweight convolution methods, GSConv, and cross-stage partial network module VoV-GSCSP as replacements for the original Conv and C3 modules in the feature fusion network. The network structures of GSConv and VoVGSCSP modules are illustrated in Figure 5 with the aim of alleviating the trade-off between model efficiency and accuracy, particularly in handling large-sized targets and optimizing the overall performance of the YOLOv5s model.

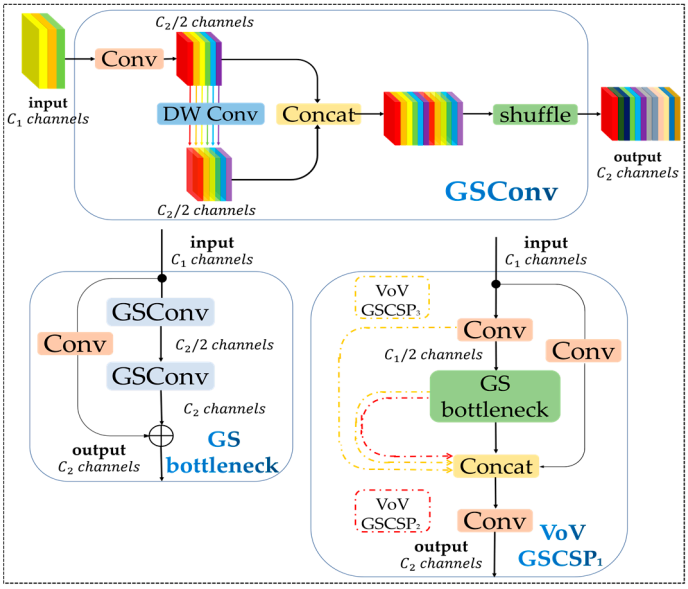


Figure 5. Network architecture of the GSConv and VoVGSCSP modules.

2.4.3. Global Attention Mechanism (GAM)

During the training of convolutional neural networks, it is essential to prioritize and select the "importance" of inputs. The attention mechanism is a method that allows the neural network to adaptively focus on specific parts internally [23]. To enhance the extraction of target features, the GAM is incorporated into the feature extraction network. This attention mechanism adopts the sequential channel–spatial attention mechanism from CBAM with a redesign of its submodules.

In apple recognition tasks conducted in natural environments, challenges such as fruit overlap, branch occlusion, and lighting variations can result in the loss of global feature information for target apples. By incorporating the GAM, which integrates both

channel and spatial attention mechanisms, attention weights are assigned to inputs that exhibit a higher prevalence of challenges such as overlapping, occlusion, and shadow challenges in the training task. The channel attention module (CAM) emphasizes color and texture feature weights, while the spatial attention module (SAM) prioritizes spatial position feature weights, improving the accuracy of apple recognition and classification in natural environments.

As shown in Figure 6, given the input feature map $F_1 \in R^{C*H*W}$, the intermediate state F_2, and the output F_3 are defined as follows:

$$F_2 = M_C(F_1) \bigotimes F_1 \tag{1}$$

$$F_3 = M_S(F_2) \bigotimes F_2 \tag{2}$$

Here, M_C and M_S represent the channel and spatial attention modules, and $\otimes$ represents element-wise multiplication.

In detail, the CAM takes a c × (w × h) feature map F_1. It first extracts the feature values of each channel through a global average pooling layer, obtaining a vector of the same length as the number of channels in the feature map. Then, using a multi-layer perceptron (MLP), this channel vector is transformed into channel weights, resulting in the weight vector M_C. Finally, the weight vector is element-wise multiplied with the input feature map to produce the weighted feature map $M_C(F_1)$. The SAM takes a c × (w × h) feature map F_2. It employs two convolutional layers for spatial information fusion, reducing and increasing the channel dimension (c) with a compression rate (r). Two feature maps are obtained and undergo channel-based feature fusion. After a sigmoid activation operation, the new feature map $M_S(F_2)$ is produced through the SAM module.

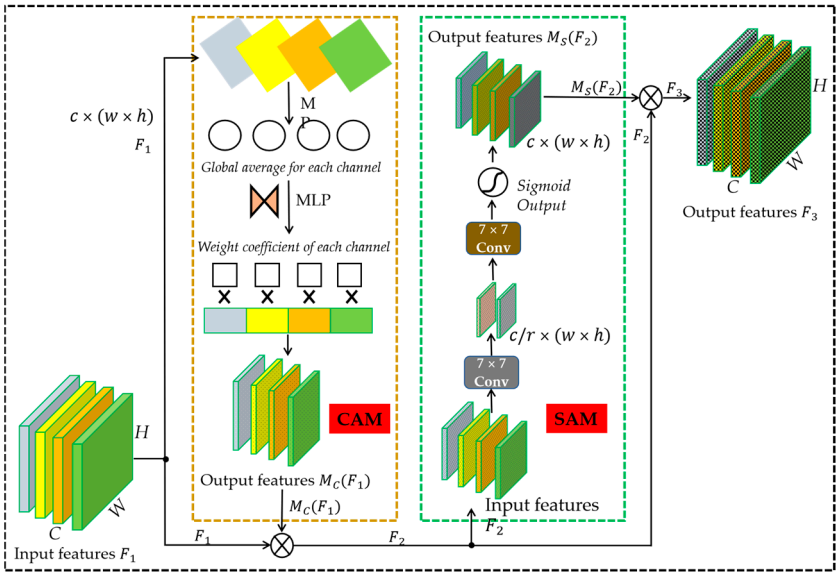


Figure 6. Network architecture of the GAM modules.

2.5. Depth Image-Assisted Apple Diameter Measurement Algorithm

Fruit diameter is a crucial parameter for grading apples, and, according to the national standard GB/T 10651-2008 for fresh apples, each grade has specific requirements for diameter, allowing a 5% tolerance above or below the specified range. Therefore, precise and rapid measurement of apple diameter is indispensable for automated apple harvesting. In this study, the Intel D457 depth camera was utilized to capture both depth and color images, with the image dimensions set at 640 × 480 pixels.

Under natural lighting conditions, the color images captured by the camera may be prone to distortions, and, simultaneously, depth data may experience information loss due to lighting variations and sensor noise [24]. To ensure accurate and swift measurement of

apple diameter, primary attention was given to correcting distortions in the color images and repairing depth data. Current conventional depth filtering methods, such as Gaussian filtering, median filtering, temporal filtering, spatial filtering, and edge-preserving filtering, often necessitate whole-image processing, resulting in slower processing speeds [25]. Considering our exclusive focus on the depth data of the detected apple target area, we adopted a localized approach that emphasized the removal and completion of depth data in the identified fruit target region. The specific processing procedure is illustrated in Figure 7, depicting the workflow for depth data processing in a singular target region. This localized depth data processing method proved more efficient in enhancing processing speed while maintaining attention to the apple target area [26].

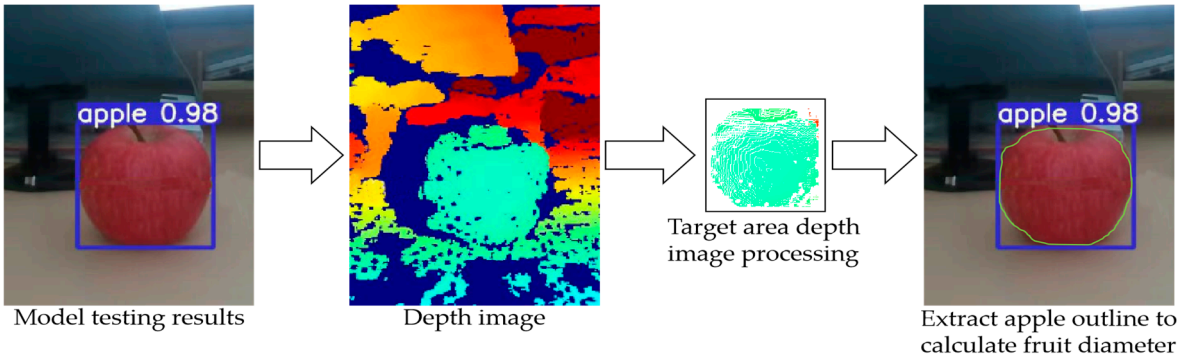


Figure 7. Apple contour extraction method process flow.

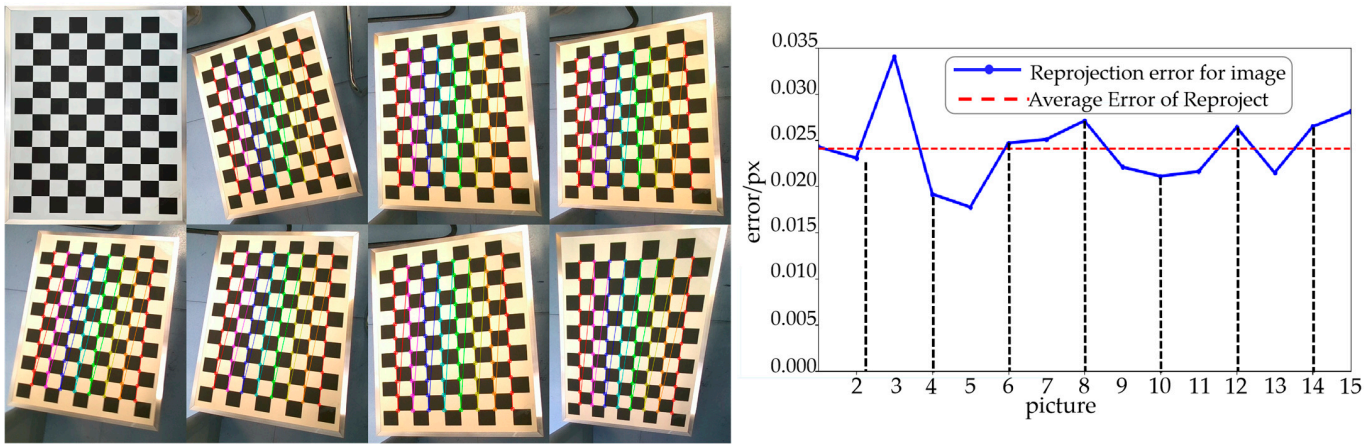
In this study, the chessboard calibration method was employed utilizing a high-precision aluminum base with a 12 × 9 chessboard pattern. As the Figure 8a shows, 30 images of the chessboard pattern were captured at different angles. Leveraging the OpenCV library, we successfully extracted the chessboard corners from these images, facilitating the calibration of the camera. The calibration process yielded the intrinsic matrix and distortion parameters of the camera, with the calibrated intrinsic matrix represented as follows:

$$K = \begin{pmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} 644.985 & 0 & 432.072 \\ 0 & 674.731 & 226.908 \\ 0 & 0 & 1 \end{pmatrix} \quad (3)$$

To assess the quality of images before and after distortion correction, we introduced the concept of a reprojection error. A reprojection error measures the deviation between points on the reconstructed image using the obtained calibration matrices and distortion parameters and the detected points on the original image. In this study, 15 randomly selected calibrated chessboard pattern images were used to calculate the reprojection error, yielding an average reprojection error of approximately 0.0249, as illustrated in Figure 8b.

For accurate apple diameter measurement, it is crucial to determine the pixel coordinates of the detected fruit target's center point and contour [27]. Considering the inaccuracies in defining the boundaries of the output detection boxes and the limited accuracy of conventional color-based segmentation algorithms when dealing with apples exhibiting significant color variations simultaneously, this paper proposes a depth-image-assisted apple contour extraction algorithm.

The algorithm initially aligns the depth map with the color map. Subsequently, the depth sub-map is generated by cropping the depth map using the pixel coordinates of the initial output bounding box obtained from the detection network and specifically targeting the region containing the fruit. This is followed by a process of restoring and refining the depth data within this sub-map. The cv2.applyColorMap() function is applied to the depth map for color mapping of depth values, facilitating a visual emphasis on the depth information related to the fruit. To ensure clearer mapping results, a scaling factor of 0.008 was set for depth value mapping, mapping the depth value range to the 0 to 255 range.



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

Figure 8. (**a**) Chessboard corners calibration; (**b**) reprojection error curve.

Considering that the maximum radius of an apple generally does not exceed 60 mm, a depth threshold of 60 was established. The depth filtering algorithm was utilized to eliminate depth information that was not associated with the fruit surface. Points with zero and excessively high depth values in the depth sub-map were remapped to white color.

Subsequently, Otsu's method, a maximum inter-class variance algorithm, was utilized within the bounding box to determine the optimal segmentation threshold, facilitating image binarization. After grayscale image binarization, morphological closing operations were applied to fill in pixel holes within the image, ultimately extracting the contour of the fruit.

The process flow of the apple diameter calculation method is depicted in Figure 9, where 2α and 2β represent the horizontal and vertical field angles of the camera, respectively, and the resolution of the camera is $W \times H$. D represents the depth of the target in the image. First, we need to calculate the conversion relationship between the pixel area and the actual area at a specific depth, as follows:

The actual width of a pixel in the horizontal direction, L_W:

$$L_W = \frac{2 \times D \times tan\ \alpha}{W} \tag{4}$$

The actual height of a pixel in the vertical direction, L_H:

$$L_H = \frac{2 \times D \times tan\ \beta}{H} \tag{5}$$

The actual area corresponding to each pixel at the target depth,S_L:

$$S_L = \frac{4 \times D^2 \times tan\ \alpha \times tan\ \beta}{WH} \tag{6}$$

Next, by fitting the depth of the circular apple contour obtained from the depth camera combined with the pixel area S_P within the contour, the final calculation formula for the apple diameter R is as follows:

$$R = \frac{4 \times D^2 \times tan\ \alpha \times tan\ \beta \times S_P}{WH\pi} \tag{7}$$

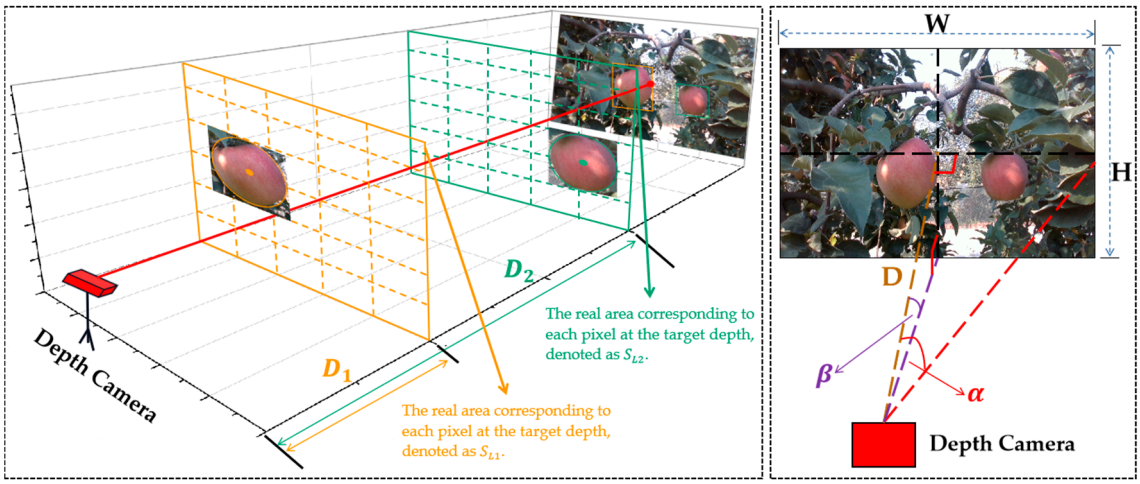


Figure 9. Principle diagram of diameter calculation method.

3. Model Training and Improvement Analysis

3.1. Model Evaluation

This study employed accuracy (P), recall (R), average precision (AP), and mean average precision (mAP) as evaluation metrics for model detection accuracy [28]. Accuracy (P) measures the ratio of correctly predicted positive observations to the total predicted positives. Recall (R) calculates the ratio of correctly predicted positive observations to all observations in the actual class. Average precision (AP) assesses the model's detection capability for each class and represents the average precision per class. Mean average precision (mAP) provides the average precision across all classes, evaluating the overall detection capability of the model.

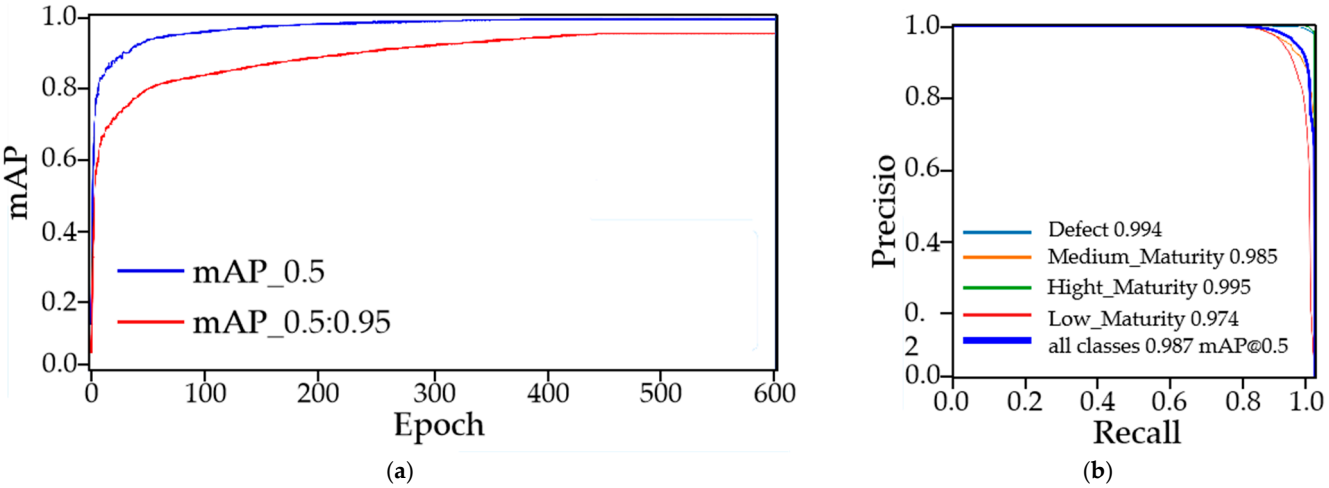
Additionally, the performance of the model was assessed using the metrics of frames per second (Fps) and parameters (Params). Fps indicates the number of images detected per second, providing insights into real-time processing capabilities. Parameters quantify the network complexity and model size.

3.2. Model Training

The network model employed in this study was trained on the Windows 10 operating system utilizing the PyTorch framework. All algorithms underwent training in a consistent environment initialized with the subsequent parameters: the image input size (img-size) was set to 640 × 640, the batch size (batch-size) for each training iteration was eight, and the number of epochs (epoch) for dataset iteration was set to 500. The initial learning rate was established at 0.01 with a momentum parameter of 0.94 and a weight decay parameter of 0.0005.

Additionally, the IOU threshold was set to 0.2, and the anchor filtering threshold was set to 4.0. In the post-processing stage, a predicted box was considered correct if its intersection over union (IOU) with the ground truth box was greater than the IOU threshold. Anchor boxes were selected only if the IOU between the anchor box and the ground truth box was greater than the anchor filtering threshold.

The change in mAP with thresholds of 0.5 and 0.5:0.95 during training is illustrated in Figure 10a. From the graph, it can be observed that, after approximately 450 epochs, both curves stabilized, indicating saturation in training. The precision–recall (P–R) curve for the best-trained model is depicted in Figure 10b, where the AP value for the fruit surface defect class was 0.994. The AP value for the medium maturity class was 0.985, for the high maturity class was 0.995, and for the low maturity class was 0.974. The mAP@0.5, representing the total average AP across all classes, was 0.987.



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

Figure 10. (**a**) mAP change during training with thresholds of 0.5 and 0.5:0.95; (**b**) precision–recall curve for the best-trained model.

3.3. Comparative Experiments on the Network Model with GSConv and VoVGSCSP Replacements

In order to compare the performance of traditional convolution (Conv) and GSConv convolution in feature map processing, we conducted comparative experiments using PyTorch. The experiments were performed on an NVIDIA GeForce RTX 3080 Ti GPU with batch size set to eight and input matrix size at 128 × 128. We tested the frames per second (FPS), floating-point operations per second (FLOPs), and the number of parameters (Params) processed by the two convolutions when handling feature information with 256 and 512 input channels. As shown in Table 3, when processing feature information with different channel numbers, GSConv convolution processes a similar amount of feature information per second as traditional convolution. However, the size of the occupied parameters is reduced by approximately half. Therefore, replacing Conv with GSConv in the network significantly reduces the model's memory footprint [29].

Table 3. Comparative experiment results for convolution types.

Index / Category	Fps ($f \cdot s^{-1}$)		FLOPs(G)		Params(K)	
	256	**512**	**256**	**512**	**256**	**512**
Conv	365.72	120.35	38.72	154.75	590.60	2417.66
GSConv	383.94	157.46	19.60	77.86	298.88	1216.51

In Figure 11, the structure diagram of the improved feature fusion network part (GS-Neck) is presented. To better understand the effect of convolutional neural networks on extracting apple feature information before and after the improvement, the visualization of feature extraction effects in some layers of the network detection layer was analyzed using the Grad-CAM heat map. Specifically, Conv and C3 layers in the original network detection layer and GSConv and VoVGSCSP layers in the improved network detection layer were selected for analysis. The visualized feature extraction effects are illustrated in Figure 12. The figures clearly demonstrate that the Conv and C3 layers in the detection layer of the original network extracted feature information ineffectively with unclear target contour edge information and indistinct extraction of key target areas influenced significantly by background noise. The improved model, through more precise feature extraction, minimized the interference of background noise on target features. As a result, there was an enhanced focus on target edge features and key areas. The heat map indicates that different categories of fruit received good differentiation in terms of texture, color, and other features.

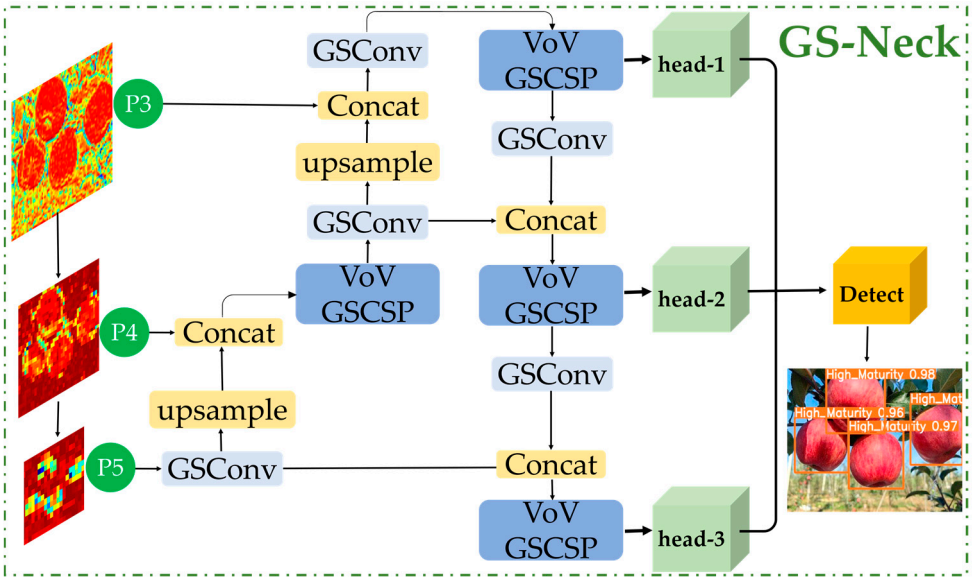


Figure 11. Network architecture of the GS-Neck.

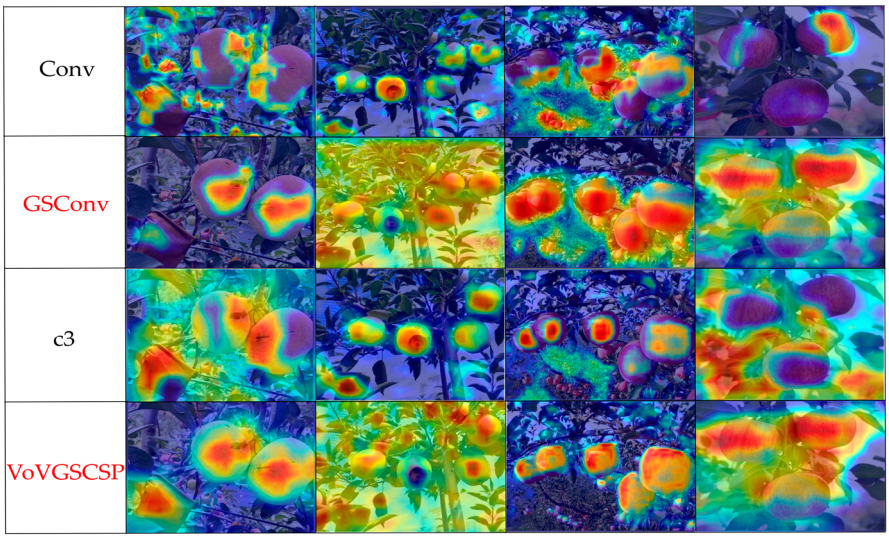


Figure 12. Visual comparison of heatmap visualization effects.

3.4. Comparative Experiments on Applying GAM Attention Mechanism at Different Positions

As depicted in Figure 13, we explored six combinations by applying the GAM attention mechanism at three different positions in the improved feature fusion network, GS-Neck. Table 4 displays the model performance for each combination. A comparative analysis reveals that applying the GAM attention mechanism at positions one, two, and three had varying effects on the model's mAP values. Specifically, applying the GAM module at positions one and two resulted in mAP improvements of 0.95% and 1.69%, respectively. Meanwhile, applying it at position three led to a 0.49% increase. Positions one and two were situated in the network at locations where multi-scale feature information was fused, allowing them to capture more comprehensive feature information compared to position three. Given this observation, we conducted experiments separately for positions one and two, revealing that applying the GAM module at these locations had the most significant impact, resulting in a 2.07% increase in mAP values. Additionally, the introduction of the GAM attention mechanism at positions one and two resulted in a manageable increase in parameters. Consequently, we selected the application of the GAM attention mechanism at positions one and two in the network as the final improvement to the model structure.

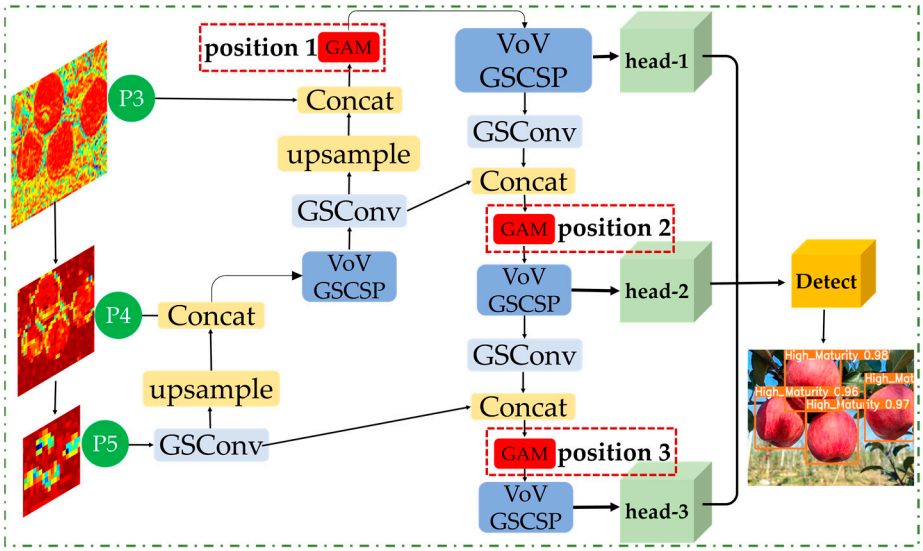


Figure 13. Application of GAM attention mechanism at three positions in the GS-Neck network.

Table 4. Model performance for each combination.

Position \ Index	mAP@0.5 (%)	Params(K)
None	96.63	11,855
Position 1	97.58	15,220
Position 2	98.32	16,486
Position 3	97.12	14,101
Position 1, 2	98.70	18,748
Position 1, 3	98.45	17,936
Position 2, 3	98.87	17,657

4. Experimental Results Analysis and Discussion

4.1. Ablation Experiments

To validate the effectiveness of the improved YOLOv5s-GGV network model for apple ripeness detection, ablation experiments were conducted. The improved components, including GSConv, VoVGSCSP, and the GAM attention mechanism, were systematically analyzed. Comparative experiments were performed against the original YOLOv5s network model. The results were summarized in Table 5.

Table 5. Comparison of model improvements experiments.

Model \ Index	AP (%)				mAP@0.5 (%)	Fps ($f \cdot s^{-1}$)	Params(K)
	Class1	Class2	Class3	Class4			
YOLOv5s	90.1	91.4	98.3	71.0	87.7	182	14,001
YOLOv5s-GAM	97.9	95.1	99.4	88.9	95.4	137	16,872
YOLOv5s-GV	98.2	95.8	99.4	93.8	96.8	191	15,423
YOLOv5s-GGV	99.4	98.5	99.5	97.4	98.7	155	18,748

In Table 5, YOLOv5s represents the unmodified original algorithm, YOLOv5s-GAM denotes the improved algorithm with the GAM attention mechanism applied at positions one and two, YOLOv5s-GV signifies the enhanced algorithm after replacing the network feature fusion component, and YOLOv5s-GGV indicates the simultaneous application of both improvements to the YOLOv5s algorithm. As shown in Table 5, applying attention mechanisms and replacing the feature fusion section in the original network significantly

improved detection accuracy. Notably, there was a significant enhancement in the detection accuracy for the low ripeness category (Low_Maturity). Figure 14 demonstrated that these two improvement methods effectively addressed the challenge of distinguishing color features between low maturity apples and tree leaves, leading to substantial increases in AP values by 17.9% and 22.8%, respectively.

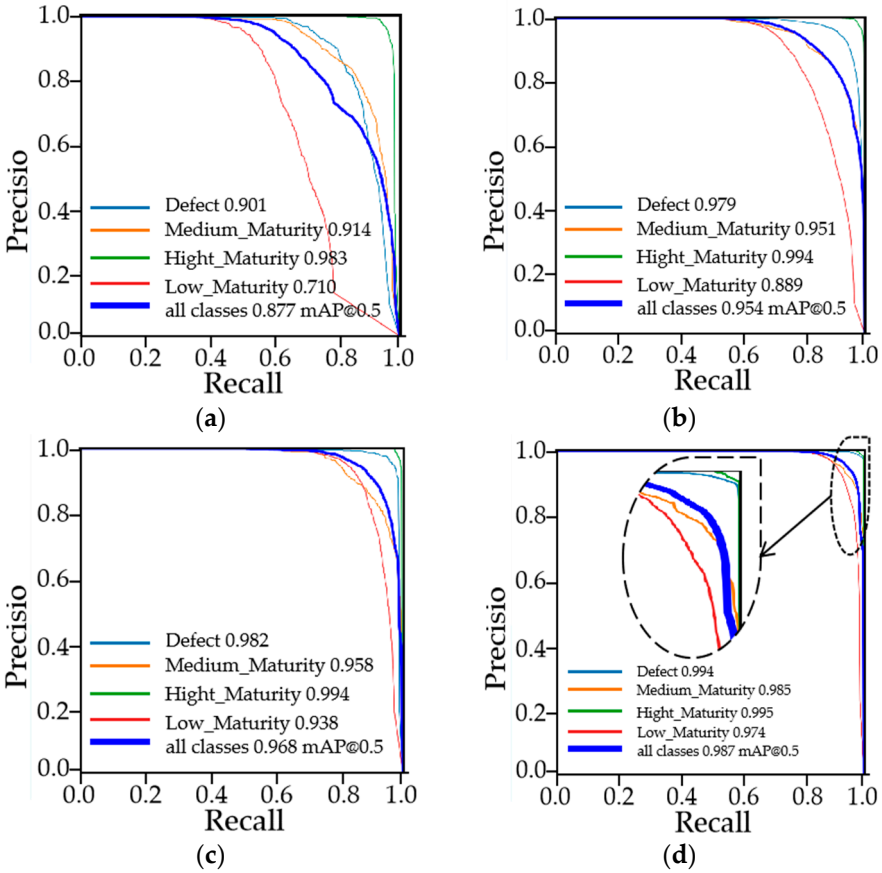


Figure 14. Comparison of model PR curve. (**a**) YOLOv5s; (**b**) YOLOv5s-GAM; (**c**) YOLOv5s-GV; (**d**) YOLOv5s-GGV.

4.2. Comparative Experiments of Different Network Detection Models

To assess the effectiveness of the improved YOLOv5s-GGV network model in apple target detection, comparative experiments were conducted with five prevalent convolutional neural networks, YOLOv5s, Faster R-CNN, Mask R-CNN, SSD512, and YOLOv7, using an orchard apple dataset. The results are presented in Table 6. Figure 15 illustrates the comparison of the original YOLOv5s network and the improved YOLOv5s-GGV network for high, medium, low maturity, and defective fruit categories. The detection results for these four categories are color-coded, with red triangles marking false positives and yellow triangles marking false negatives.

As indicated in Table 6, the improved YOLOv5s-GGV network achieved the highest overall average recognition accuracy at 98.7%. Moreover, it significantly outperformed two-stage object detection algorithms, namely Faster R-CNN and Mask R-CNN, in terms of detection speed. YOLOv5s belonged to the single-stage object detection category, boasting fast detection speeds suitable for real-time requirements. On the other hand, two-stage object detection algorithms represented by Mask R-CNN, while offering higher detection accuracy, suffered from slow detection speeds and large model parameter sizes, making them less suitable for deployment on embedded platforms. YOLOv5s-GGV, in comparison, exhibited a 5.5% increase in overall average accuracy over Mask R-CNN with a speed increase of 150 frames per second, meeting the real-time detection precision and speed

requirements for harvesting robots. While YOLOv7 showed a marginal improvement in detection speed, its accuracy decreased by 10.1% compared to YOLOv5s-GGV. With a model size of 18.1MB, YOLOv5s-GGV has significantly reduced parameters compared to two-stage object detection models, making it four times smaller than YOLOv7 in terms of model parameters. This size reduction positions YOLOv5s-GGV with a distinct advantage in detection speed on embedded platforms.

Table 6. Comparison of training results for different models.

Model \ Index	mAP@0.5 (%)	Fps (f·s^{-1})	Params(K)
YOLOv5s	87.7	182	14,001
Faster R-CNN	92.1	8	110,766
Mask R-CNN	93.2	5	249,856
SSD300	83.4	56	92,160
YOLOv7	88.6	161	73,816
YOLOv5s-GGV	98.7	155	18,748



Figure 15. Comparison of detection performance for different models. (Red triangles represent false detections by the model, indicating category detection errors; yellow triangles represent missed detections by the model, indicating failure to detect the expected targets).

4.3. Laboratory Apple Diameter Measurement Experiment

To validate the accuracy of fruit size determination after distortion removal and depth filtering, this study collected 10 apples as research subjects and measured their actual diameters. Referring to the national standard GB/T 10651-2008 for fresh apples' size classification, three diameter thresholds were established: 85 mm, 80 mm, and 75 mm. Classifications were defined as follows: apples with a diameter (R) greater than or equal to 85 mm were considered first-grade, those with 80 mm $\leq$ R < 85 mm were second-grade, 75 mm $\leq$ R < 80 mm were third-grade, and those with R < 75 mm were fourth-grade. Initially, the actual diameters of the apples used in the experiment were measured using calipers and sequentially numbered. The experiment was conducted three times with

randomized apple positions. We conducted three experiments on a randomly shuffled selection of ten apples, testing the accuracy of algorithms measuring the diameter of the same apple at different positions in the image. This was done to verify the effectiveness of image distortion correction. Figure 16 displays the colored images and detection results obtained by the camera in three experimental runs. Figure 17 illustrates the effectiveness of the algorithm in extracting fruit contours during the processing steps.

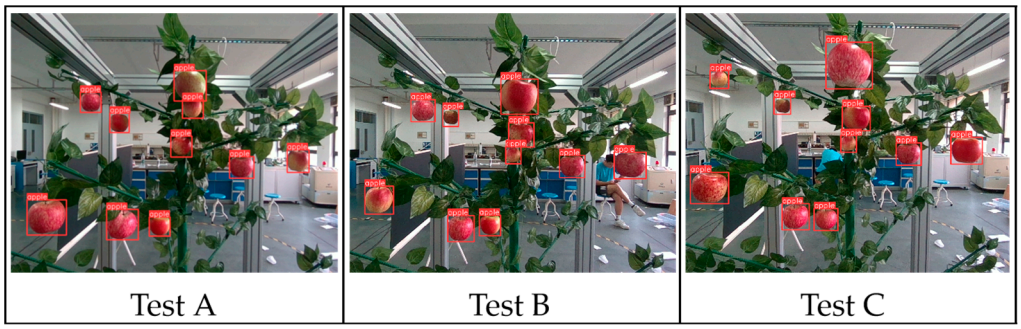


Figure 16. Color images and detection results.

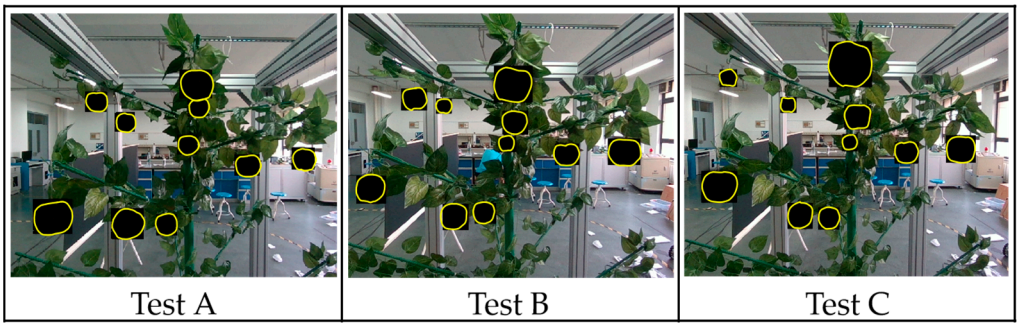


Figure 17. Apple contour extraction results.

Finally, based on the depth-assisted apple sizing algorithm introduced in Section 2.5 of this paper, the estimated apple diameters from the camera were obtained. These estimated diameters were then compared with the actual diameters of the corresponding apples and categorized, as shown in Table 7.

Table 7. Comparison of training results for different models.

ID	R	R_1(mm)			R_2(mm)		
		A(*)	B(*)	C(*)	A(*)	B(*)	C(*)
1	67.80 (4)	72.06 (4)	66.06 (4)	69.95 (4)	65.99 (4)	68.06 (4)	67.95 (4)
2	68.40 (4)	69.95 (4)	66.00 (4)	70.01 (4)	68.00 (4)	67.07 (4)	68.00 (4)
3	75.92 (3)	80.21 (2)	73.96 (4)	78.00 (3)	74.32 (4)	76.01 (3)	74.76 (3)
4	83.80 (2)	85.02 (1)	85.30 (1)	84.47 (2)	83.99 (2)	82.02 (2)	83.96 (2)
5	84.42 (2)	86.00 (1)	85.99 (1)	86.08 (1)	83.07 (2)	85.02 (1)	82.02 (2)
6	58.56 (4)	60.05 (4)	62.00 (4)	63.26 (4)	61.21 (4)	60.00 (4)	58.01 (4)
7	83.22 (2)	86.00 (1)	79.99 (2)	81.96 (2)	84.07 (2)	83.96 (2)	83.99 (2)
8	100.90 (1)	96.00 (1)	104.09 (1)	103.90 (1)	101.90 (1)	99.98 (1)	100.01 (1)
9	88.12 (1)	86.02 (1)	89.97 (1)	89.99 (1)	87.02 (1)	88.07 (1)	86.82 (1)
10	81.88 (2)	83.96 (2)	77.99 (3)	83.96 (2)	82.00 (2)	81.99 (2)	81.72 (2)

Notes: R represents the actual measured diameter; R_1 is the diameter measured by the algorithm before processing the depth map and color image; R_2 is the diameter measured by the algorithm after image processing; A$(*)$, B$(*)$, C$(*)$ represent three experiments, (*) indicates the grading result, and (*) indicates a discrepancy between the algorithm's grading result and the actual grading result.

As illustrated in Table 7, after three experiments involving the classification of 30 apples based on their diameters, the algorithm's pre-processing measurements resulted in accurate classifications for 21 apples, while nine were misclassified, yielding a success rate of 70%. Following image processing, which included distortion correction and depth filtering, the algorithm exhibited improved accuracy, correctly classifying 28 apples with only two misclassifications, achieving a success rate of 93.3%. Consequently, the post-processing of images led to a substantial 23.3% increase in classification accuracy. Figure 18 illustrates the maximum deviation between the actual diameter of apples after image processing and the algorithm-determined value is 2.65 mm, while the minimum deviation is 0.05 mm. The average deviation of the diameters of 10 apple targets in three trials is approximately 0.878 mm. Notably, the reduced absolute error after image processing underscores the enhanced accuracy in diameter measurements by the algorithm.

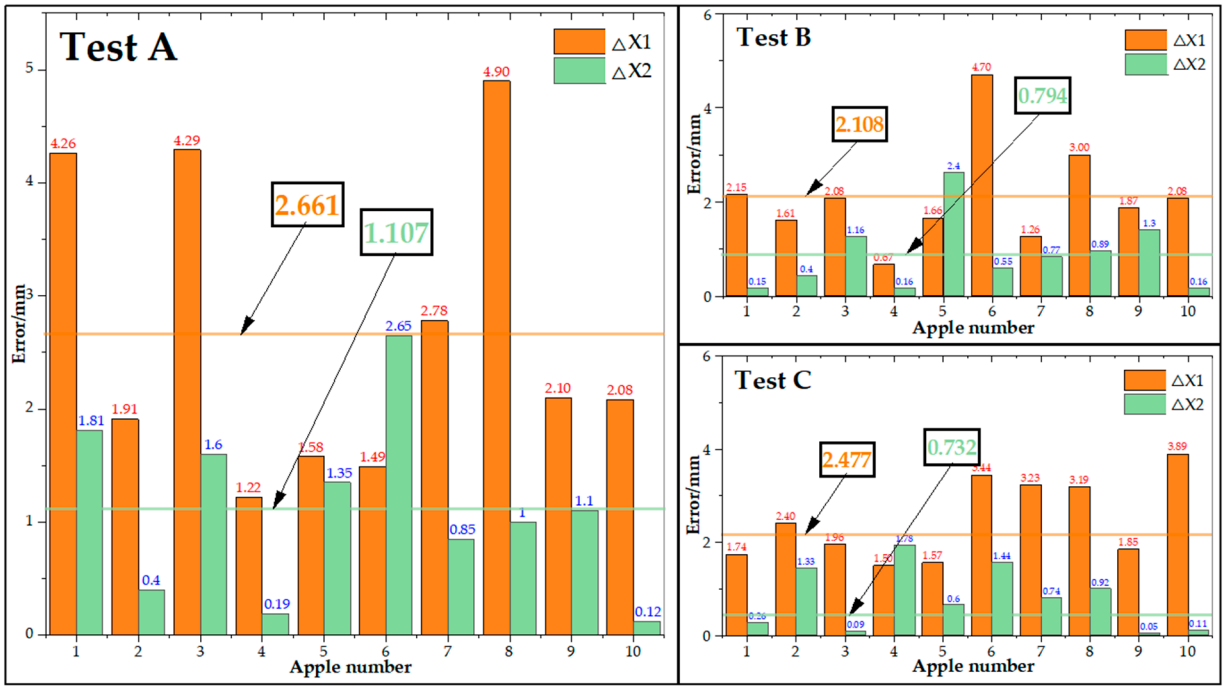


Figure 18. Absolute error in apple diameter measurement by the algorithm (ΔX1: absolute error between algorithmic and actual diameter measurements before image processing; ΔX2: absolute error between algorithmic and actual diameter measurements after image processing).

Furthermore, the algorithm's average processing time per image was a mere 0.018 s, thereby satisfying the real-time precision requirements for vision-based robotic apple harvesting. This affirmed the effectiveness of the algorithm and emphasized the crucial role of image processing in refining diameter measurement accuracy. The observed proximity between algorithmic measurements and actual diameters post-image processing not only highlights the algorithm's superiority in diameter feature detection but also underscores the pivotal contribution of image processing in elevating measurement precision. In conclusion, the algorithm demonstrated not only efficiency in real-time vision for apple-picking robots but also substantial improvements in accuracy through meticulous image processing.

4.4. Discussion

1. Table 8 provides a performance comparison between the YOLOv5s-GGV model proposed in this paper and several of the latest existing apple target detection models.

Table 8. Performance comparison of the proposed methods with related works.

Method	Applicable Environment	Detection Model	mAP@0.5 (%)	Fps (f·s^{-1})
Reference [30]	Multiple target overlap and foliage occlusion in the orchard	Improve YOLOv5	96.4	75.26
Reference [31]	Discrimination of different maturity apples in automatic picking	SODSTR-YOLOv5s	93.6	62.5
Reference [32]	Complex natural orchard environment	Improve YOLOv7	97.0	75.85
This research	Discrimination of different maturity apples in automatic picking	YOLOv5s-GGV	98.7	155

The annotation of the dataset in this paper primarily aims to identify fully harvestable apples for the robotic arm without considering occlusions. As shown in Table 8, the proposed YOLOv5s-GGV model also achieved the highest detection accuracy and speed. Compared with the method presented in reference [31] within the same application environment, there was an improvement of 5.1% in detection accuracy and an increase of 92.5 frames per second (fps) in detection speed. Considering the model performance in comparative literature comprehensively, the YOLOv5s-GGV model proposed in this paper exhibits more significant advantages. It meets the real-time detection requirements of harvesting robots while ensuring detection accuracy.

2. In reference [26], Mar Ferrer-Ferrer and colleagues introduced a method for simultaneous apple detection and size estimation using a multi-task deep neural network. Their research indicated that the average absolute error in diameter estimation was 5.09 mm. However, in this paper, we present a novel method for fruit diameter measurement, which has an average absolute error of only 0.878 mm, demonstrating a significantly more accurate performance in comparison.

5. Conclusions

This study proposes an online detection method for assessing the ripeness and diameter features of Fuji apples based on YOLOv5s. This approach, tailored to the real-time demands of apple-picking robots, maintained high detection accuracy. Experimental validations were conducted separately to evaluate the ripeness detection and diameter estimation capabilities of the proposed online detection method. The primary conclusions derived from this research are outlined as follows:

1. Through comparative experiments between traditional convolution (Conv) and GSConv convolution, it was observed that GSConv convolution demonstrated comparable performance to Conv while reducing parameters by approximately half during frame processing. In the YOLOv5s network model, structural optimizations and Grad-CAM visualization revealed that the improved model excelled in accurately and clearly extracting target features. The incorporation of the GAM attention mechanism into the GSConv foundation, applied at various network model positions, resulted in significant performance enhancement, particularly with a 2.07% increase in mAP values when the GAM module was added at positions one and two. Further ablation experiments combining GSConv, VoVGSCSP, and GAM formed the YOLOv5s-GGV model, exhibiting substantial improvements, notably in the low maturity category, where AP values increased by 17.9% and 22.8%.

When comparing different network detection models, YOLOv5s-GGV achieved the highest overall average recognition accuracy of 98.7% on the orchard apple dataset, outperforming Faster R-CNN and Mask R-CNN while meeting real-time requirements regarding detection speed. In conclusion, the enhanced YOLOv5s-GGV model excelled in apple ripeness detection, providing higher detection accuracy and faster speed, thus offering an effective solution for real-time detection and classification in robotic harvesting.

2. In order to assess the impact of distortion removal and depth filtering on the accuracy of fruit size determination, laboratory experiments were conducted on apple diameter measurement. Ten apples were selected as study subjects, and their real diameters were

measured following the national standard GB/T 10651-2008 for apple size classification. Three diameter thresholds were set. Initially, the actual diameters were measured using a vernier caliper, and, subsequently, the apples were randomly suspended on a simulated apple tree to capture color and depth images before and after distortion removal as well as depth filtering. Subsequently, a depth-assisted apple sizing algorithm estimated apple diameters, which were then compared with real diameters for classification.

The experimental results indicated that, prior to image processing, the algorithm correctly classified 21 apples with nine misclassifications, resulting in a success rate of 70%. However, after applying distortion removal and depth filtering techniques, the algorithm demonstrated significant improvement with a success rate of 93.3%, correctly classifying 28 apples with only two misclassifications, representing a 23.3% increase in classification accuracy. Additionally, the algorithm processed a single image in an average time of only 0.018 s, meeting the real-time and precision requirements for apple-picking robots and confirming the algorithm's efficacy. This highlights a significant performance improvement in apple diameter measurement through distortion removal and depth filtering.

Author Contributions: Writing—original draft preparation, J.L., S.L. and J.W.; writing—review and editing, J.L. and S.L. conceptualization, J.L., S.L. and J.W.; methodology, J.L., H.Y., Y.Y. and G.Z.; software, J.L., S.L., Y.L., J.S. and G.F.; validation, J.L., H.Z., H.Y. and J.W.; resources, J.W.; data curation, J.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by China Agriculture Research System (CARS-27), Shandong Province Key Research and Development Plan (2022CXGC020701) and Shandong Province Rural Revitalization Innovation Boosting Action Plan.

Data Availability Statement: The data are available within the article.

Acknowledgments: Thanks to all the authors cited in this article and the referee for their helpful comments and suggestions.

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

References

1. Kumar, P.A.; Zhang, Z.; Lu, R. Evaluation of a new apple in-field sorting system for fruit singulation, rotation and imaging. *Comput. Electron. Agric.* **2023**, *208*, 107789. [CrossRef]
2. Bhargava, A.; Bansal, A. Classification and grading of multiple varieties of apple fruit. *Food Anal. Methods* **2021**, *14*, 1359–1368. [CrossRef]
3. Zhang, Z.; Lu, Y.Z.; Lu, R.F. Development and evaluation of an apple infield grading and sorting system. Postharvest Biol. *Technol.* **2021**, *180*, 111588.
4. Liang, X.T.; Jia, X.Y.; Huang, W.Q.; He, X.; Li, L.J.; Fan, S.X.; Li, J.B.; Zhao, C.J.; Zhang, C. Real-Time Grading of Defect Apples Using Semantic Segmentation Combination with a Pruned POLO V4 Network. *Foods* **2022**, *11*, 3150. [CrossRef] [PubMed]
5. Goncalves, M.W.; Argenta, L.C.; Martin, M.S.D. Maturity and quality of apple fruit during the harvest period at apple industry. *Rev. Bras. Frutic.* **2017**, *39*, e-825. [CrossRef]
6. Sousa, M.L.; Gonçalves, M.; Fialho, D.; Ramos, A.; Lopes, J.P.; Oliveira, C.M.; De Melo-Abreu, J.P. Apple and Pear Model for Optimal Production and Fruit Grade in a Changing Environment. *Horticulturae* **2022**, *8*, 873. [CrossRef]
7. Nie, W.J.; Abler, D.; Li, T.P. Grading attribute selection of China's grading system for agricultural products: What attributes benefit consumers more? *J. Behav. Exp. Econ.* **2021**, *13*, 2167. [CrossRef]
8. Lu, Y.Z.; Lu, R.F.; Zhang, Z. Development and preliminary evaluation of a new apple harvest assist and in-field sorting machine. *Appl. Eng. Agric.* **2022**, *38*, 23–35. [CrossRef]
9. Geng, X.; Zhang, J.; Liang, X.; Yang, Y.; Zhang, W. Design and implementation of Red Fuji apple online grading system. *Electron. Des. Eng.* **2023**, *31*, 124–128.
10. Yan, J.; Zhao, Y.; Zhang, L.; Su, X.; Liu, H.; Zhang, F.; Fan, W.; He, L. Recognition of Rosa roxbunghii in natural environment based on improved Faster RCNN. *Trans. Chin. Soc. Agric. Eng.* **2019**, *35*, 143–150.
11. Cubero, S.; Aleixos, N.; Albert, F.; Torregrosa, A.; Ortiz, C.; García-Navarrete, O.; Blasco, J. Optimised computer vision system for automatic pre-grading of citrus fruit in the field using a mobile platform. *Precis. Agric.* **2014**, *15*, 80–94. [CrossRef]
12. Moallem, P.; Serajoddin, A.; Pourghassem, H. Computer vision-based apple grading for golden delicious apples based on surface features. *Inf. Process. Agric.* **2017**, *4*, 33–40. [CrossRef]
13. Hu, G.; Zhang, E.; Zhou, J.; Zhao, J.; Gao, Z.; Sugirbay, A.; Chen, J. Infield apple detection and grading based on multi-feature fusion. *Horticulturae* **2021**, *7*, 276. [CrossRef]

14. Shi, X.; Chai, X.; Yang, C.; Xia, X.; Sun, T. Vision-based apple quality grading with multi-view spatial network. *Comput. Electron. Agric.* **2022**, *195*, 106793. [CrossRef]
15. Montoya-Cavero, L.E.; de León Torres, R.D.; Gómez-Espinosa, A.; Cabello, J.A.E. Vision systems for harvesting robots: Produce detection and localization. *Comput. Electron. Agric.* **2022**, *192*, 106562. [CrossRef]
16. Wang, D.; He, D. Fusion of Mask RCNN and attention mechanism for instance segmentation of apples under complex background. *Comput. Electron. Agric.* **2022**, *196*, 106864. [CrossRef]
17. Zhang, L.; Hao, Q.; Cao, J. Attention-Based Fine-Grained Lightweight Architecture for Fuji Apple Maturity Classification in an Open-World Orchard Environment. *Agriculture* **2023**, *13*, 228. [CrossRef]
18. Tian, Y.; Yang, G.; Wang, Z.; Wang, H.; Li, E.; Liang, Z. Apple detection during different growth stages in orchards using the improved YOLO-V3 model. *Comput. Electron. Agric.* **2019**, *157*, 417–426. [CrossRef]
19. Lou, J.; Wang, M. Research on Fruit Quality Detection and Classification Method Based on YOLOv5. *Softw. Guide* **2023**, *22*, 190–195.
20. Li, Z.; Song, Y.; Xu, R.; Li, F.; Zheng, G. Fruit volume measurement algorithms based on SFM and deep learning. *Comput. Eng. Des.* **2023**, *44*, 1699–1705.
21. Ratha, A.K.; Barpanda, N.K.; Sethy, P.K.; Sharada, G.; Behera, S.K. Computer Intelligence-Based Fruit Grading: A Review. *Rev. D'intelligence Artif.* **2023**, *37*, 465–474. [CrossRef]
22. Li, H.; Li, J.; Wei, H.; Liu, Z.; Zhan, Z.; Ren, Q. Slim-neck by GSConv: A better design paradigm of detector architectures for autonomous vehicles. *arXiv* **2022**, arXiv:2206.02424.
23. Liu, Y.; Shao, Z.; Hoffmann, N. Global attention mechanism: Retain information to enhance channel-spatial interactions. *arXiv* **2021**, arXiv:2112.05561.
24. Zhang, L.; Xia, H.; Qiao, Y. Texture synthesis repair of RealSense D435i depth images with object-oriented RGB image Segmentation. *Sensors* **2020**, *20*, 6725. [CrossRef] [PubMed]
25. Chang, T.A.; Yang, J.F. Precise depth map upsampling and enhancement based on edge-preserving fusion filters. *IET Comput. Vis.* **2018**, *12*, 651–658. [CrossRef]
26. Ferrer-Ferrer, M.; Ruiz-Hidalgo, J.; Gregorio, E.; Vilaplana, V.; Morros, J.R.; Gené-Mola, J. Simultaneous fruit detection and size estimation using multitask deep neural networks. *Biosyst. Eng.* **2023**, *233*, 63–75. [CrossRef]
27. Yue, L.X.; Li, W.K.; Yang, X.F.; Li, H.F.; Yang, Q.S. Apple Detection and Fruit Diameter Estimation Method Based on Improved YOLOv4. *Laser J.* **2022**, *2*, 58–65.
28. Sun, L.; Hu, G.; Chen, C.; Cai, H.; Li, C.; Zhang, S.; Chen, J. Lightweight Apple Detection in Complex Orchards Using YOLOV5-PRE. *Horticulturae* **2022**, *8*, 1169. [CrossRef]
29. He, Y.; Tian, J.W.; Zhang, Z.; Wang, Q.; Zhao, P. Lightweight Research of YOLOv5 Target Detection. *Comput. Eng. Appl.* **2023**, *59*, 92–99.
30. Kou, L.L.; Zhang, H.N. Research on multi-target recognition technology of apple picking robot based on improved YOLOv5. *J. Chin. Agric. Mech.* **2023**, *44*, 162–168.
31. Wang, Y.; Tao, Z.; Shi, X.Y.; Wu, Y.; Wu, H. Apple target detection method with different ripeness based on improved YOLOv5s. *J. Nanjing Agric. Univ.* **2024**, 1–13.
32. Zhang, Z.; Zhou, J.; Jiang, Z.; Han, H. Lightweight Apple Recognition Method in Natural Orchard Environment Based on Improved YOLOv7 Model. *Trans. Chin. Soc. Agric. Mach.* **2024**, *0924004*, 1–13.

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.