horticulturae

MDPI

Article

Location Research and Picking Experiment of an Apple-Picking Robot Based on Improved Mask R-CNN and Binocular Vision

Tianzhong Fang, Wei Chen * and Lu Han

College of Automation, Jiangsu University of Science and Technology, Zhenjiang 212100, China; 231210301205@stu.just.edu.cn (T.F.); 241110302101@stu.just.edu.cn (L.H.)
* Correspondence: cw1@just.edu.cn

Abstract

With the advancement of agricultural automation technologies, apple-harvesting robots have gradually become a focus of research. As their "perceptual core," machine vision systems directly determine picking success rates and operational efficiency. However, existing vision systems still exhibit significant shortcomings in target detection and positioning accuracy in complex orchard environments (e.g., uneven illumination, foliage occlusion, and fruit overlap), which hinders practical applications. This study proposes a visual system for apple-harvesting robots based on improved Mask R-CNN and binocular vision to achieve more precise fruit positioning. The binocular camera (ZED2i) carried by the robot acquires dual-channel apple images. An improved Mask R-CNN is employed to implement instance segmentation of apple targets in binocular images, followed by a template-matching algorithm with parallel epipolar constraints for stereo matching. Four pairs of feature points from corresponding apples in binocular images are selected to calculate disparity and depth. Experimental results demonstrate average coefficients of variation and positioning accuracy of 5.09% and 99.61%, respectively, in binocular positioning. During harvesting operations with a self-designed apple-picking robot, the single-image processing time was 0.36 s, the average single harvesting cycle duration reached 7.7 s, and the comprehensive harvesting success rate achieved 94.3%. This work presents a novel high-precision visual positioning method for apple-harvesting robots.

Keywords: apple-picking robot; binocular positioning; stereo matching; instance segmentation

check for updates

Academic Editor: Georgios Koubouris

Received: 21 May 2025
Revised: 22 June 2025
Accepted: 3 July 2025
Published: 6 July 2025

Citation: Fang, T.; Chen, W.; Han, L. Location Research and Picking Experiment of an Apple-Picking Robot Based on Improved Mask R-CNN and Binocular Vision. *Horticulturae* **2025**, *11*, 801. https://doi.org/10.3390/horticulturae11070801

Copyright: © 2025 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

According to the United Nations Food and Agriculture Organization (FAO), global apple production has continued to grow over the past decade, reaching 82.934 million tons in 2022 [1]. However, apple harvesting faces significant challenges due to declining rural labor forces. In China, the production cost per hectare of apple orchards is approximately CNY 7672, with labor costs accounting for approximately CNY 4030, representing 52.5% of the total expenditure [2]. Manual apple harvest remains a laborious and costly process, particularly when orchards contain large numbers of trees that require to be harvested. This problem is exacerbated by rising labor costs and a shrinking pool of skilled workers [3]. Therefore, there is an urgent need to investigate apple-harvesting robots to achieve efficient and cost-effective fruit picking. A harvesting robot typically comprises two primary subsystems: a vision system and an end effector system [4]. The vision system guides the end effector of the robot to detect and locate apples for precise picking from trees [5].

In recent years, foundation models and generative AI have shown remarkable progress in robotic control, exemplified by approaches like Robotics Transformers (RT-X) that generate motor actions directly from visual inputs [6] and Physical Intelligence's pi-series for embodied agents [7]. However, these data-intensive frameworks face significant deployment challenges in agricultural settings. Limited computational resources, sparse task-specific training data, and stringent real-time requirements in orchard environments necessitate more streamlined solutions. Recent studies [8,9] highlight this gap, indicating that lightweight, domain-optimized perception modules remain critical for agricultural robotics. Intelligent machine vision-based detection technologies have significantly improved the recognition accuracy and positioning efficiency of agricultural harvesting robots for apple targets. Regarding the optimization of visual positioning systems, Zhao et al. implemented fruit spatial positioning using a CCD-based monocular color vision system, with experiments demonstrating a picking success rate of 77% [10]. Abeyrathna et al. developed a composite perception system integrating an RGB-D camera with a single-point laser sensor, achieving positioning accuracy within ±2 cm error margins (excluding occluded targets) [11]. In the field of deep learning applications, the fruit detection system developed by Zhang's team exhibited remarkable time-efficiency advantages, requiring only 0.3 s to complete the recognition and positioning of all fruits in a single image [12]. Gené-Mola et al. innovatively employed mobile terrestrial laser scanning (MTLS) technology to acquire 3D point cloud data and constructed a detection algorithm combining apparent reflectance features, achieving an F1-score of 0.86 in complex scenarios [13]. To address the challenge of occluded target recognition, Yuan et al. proposed an adaptive radius selection strategy integrated with the Random Sample Consensus (RANSAC) algorithm, which effectively improved the detection accuracy of overlapping fruits while maintaining an average processing time below 50 ms per frame [14]. These research achievements provide reliable technical support for automated apple harvesting.

Binocular positioning has emerged as a widely adopted positioning method, offering biologically inspired principles, broader applicability, and lower hardware costs compared to other vision-based positioning approaches [15]. While this method is inherently constrained by its heavy reliance on feature matching, which may be susceptible to challenges such as textureless surfaces [16], such limitations rarely occur under typical agricultural harvesting robot operational conditions with natural illumination and can thus be considered negligible. Binocular vision demonstrates favorable performance in efficiency and accuracy while maintaining a simple system architecture, making it highly suitable for target recognition and positioning tasks [17]. Consequently, integrating binocular positioning into agricultural robots' perception systems represents a viable solution. Seminal work by Takahashi et al. pioneered the application of binocular stereo vision for apple harvesting in fruit recognition research, establishing a foundational framework for subsequent studies [18]. Williams et al. designed a quadrupedal kiwifruit-harvesting robot employing binocular vision, achieving a visual recognition success rates of 76.3–89.6% [19]. Furthermore, Lei et al., Luo et al., Yu et al., and Zhao et al. have successfully employed binocular positioning to obtain fruit 3D coordinates, collectively confirming the suitability of binocular vision for agricultural harvesting robots [20–23].

Prior to apple positioning, harvesting robots must first identify apples within binocular images. Common techniques for separating apples from backgrounds include object detection, semantic segmentation, and instance segmentation. However, conventional detection and segmentation methods often lack robustness in most orchard scenarios. Henila et al. proposed a Fuzzy Cluster-Based Thresholding (FCBT) method for apple fruit sorting [24]. Ibarra et al. developed a Color Dominance-Based Polynomial Optimization Segmentation approach, primarily designed to identify leaves and fruits on tomato

plants [25]. Fu et al. created an image processing algorithm leveraging the color and shape features of kiwifruits and their calyxes to separate linearly clustered fruits [26]. Mizushima and Lu introduced segmentation methods combining support vector machines with Otsu's algorithm [27–29]. While traditional image processing algorithms are computationally efficient to implement, their detection and segmentation accuracy becomes susceptible to variations in fruit/surface coloration, illumination conditions, camera perspectives, and camera-to-fruit distances [30,31]. The heterogeneous characteristics of orchard fruits and complex background variations impose heightened demands on image detection and segmentation technologies.

Compared to traditional image processing algorithms relying on manual feature design, target detection methods based on Convolutional Neural Networks (CNNs) have significantly enhanced object recognition performance in complex agricultural scenarios through autonomous learning mechanisms and multi-level feature fusion advantages [32,33]. With the advancement of deep learning technologies, researchers have developed various network models for crop target detection and semantic segmentation tasks [33–39]. However, semantic segmentation techniques still suffer from inherent limitations in intra-class object differentiation capabilities [40]. To address this, instance segmentation methods capable of distinguishing individual differences have emerged, providing refined solutions for overlapping target recognition through pixel-level positioning and instance boundary delineation. In apple-harvesting applications, Gené-Mola et al. developed a Fuji apple segmentation model based on the Mask R-CNN framework, achieving synchronized optimization with 0.86 accuracy and F1-score on 2D RGB images [41]. The Wang team innovatively implemented a DeepSnake architecture for apple detection systems, elevating the comprehensive recognition accuracy to 95.66% [42]. Kang et al. integrated a real-time instance segmentation model with a harvesting robot system, ultimately achieving an over 80% harvesting success rate [43]. Notably, instance segmentation not only provides pixel-level descriptions of target contours but also effectively resolves positioning challenges for overlapping fruits through inter-instance differentiation mechanisms, demonstrating superior engineering applicability in automated orchard harvesting operations.

Regarding the harvesting system design for apple-picking robots, Huang et al. achieved 76.97% harvesting success at 7.29 s/fruit using dual-arm vision and genetic algorithms, improving efficiency by 150% over single-arm systems [44]. Xie et al. reduced the operation time by 3.1% in 50-target scenarios via deep reinforcement learning with self-attention for spatial constraints [45]. Chen et al. attained 91.2% grasp success (3 pp higher than PID) using FNN-SMC control to suppress joint vibration [46]. Zhang et al. developed a nonlinear motion control robotic arm, integrated with an air-electric hybrid vacuum end-effector, to achieve the smart separation of apples [12]. Zhang et al. proposed a novel perception algorithm tailored for apple picking, enabling rapid identification and precise localization of apples. He also developed a four-degree-of-freedom (4-DOF) robotic arm and a flexible vacuum end-effector [47].

Although significant progress has been made in fruit-picking robots, current vision-based localization methods often face a trade-off between high accuracy and cost-effectiveness. To address these challenges, this study proposes a visual localization method for apple-picking robots that integrates an improved Mask R-CNN with binocular vision. The Intersection over Union (IoU) for apple detection and segmentation using the improved Mask R-CNN was calculated. Additionally, the Coefficient of Variation (CoV) and Positioning Accuracy (PA) were computed to evaluate the localization performance of the proposed method. Furthermore, 70 picking experiments designed for varying occlusion levels were conducted using a custom-developed apple-picking robot to validate the method's effectiveness. This research not only provides a novel solution for intelligent fruit picking but

also offers valuable insights for the development of low-cost, high-performance robotic vision systems in agricultural applications.

2. Materials and Methods

2.1. Image Acquisition

The experimental images were captured using a ZED2i stereo camera (Stereolabs, San Diego, CA, USA). The ZED2i has two sensors (Figure 1). A total of 1000 apple images were acquired under varying conditions, including different time periods (morning, afternoon, and night), diverse illumination intensities (sunny and cloudy conditions), as well as varying degrees of overlap, occlusion, and oscillation interference. The images were stored in PNG format with a resolution of 1280 × 960.



Figure 1. ZED2i binocular camera.

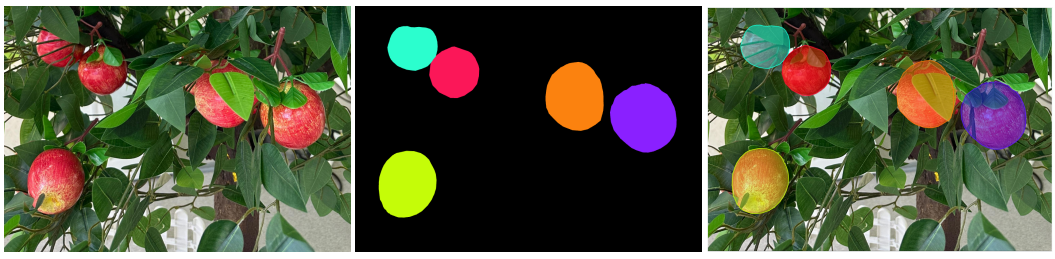
To avoid singularity in the image samples and ensure the dataset encompasses apple images under diverse natural conditions (see Figure 2), 900 images were randomly selected for training and parameter optimization of the Mask R-CNN model, with 90% allocated to the training set and 10% to the validation set. After training, the remaining 100 images were utilized for testing to evaluate the performance of the trained model.



(**a**) (**b**) (**c**) (**d**)

Figure 2. Categories of apple images: (**a**) separated fruit. (**b**) overlapping fruit. (**c**) occluded fruit. (**d**) fruit under insufficient illumination.

The experimental data were annotated using the image annotation tool Labelme to generate mask images of apples. These mask images were subsequently employed to compute backpropagation losses and optimize model parameters during training. Additionally, the performance of the trained model in instance segmentation tasks was assessed by comparing annotated mask images with predicted mask results. Fruit regions in the images were labeled, while the remaining areas were designated as background by default. The annotated apple images are shown in Figure 3.



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

Figure 3. Apple dataset example of instance segmentation: (**a**) original image. (**b**) mask image of instance segmentation. (**c**) visualization of mask image.

2.2. Training Platform

The training platform included a desktop computer with an Intel Core i7-14650HX (5.20 GHz) 16-core CPU, a GeForce GTX 4060 8 GB GPU, and 16 GB of memory, running on a Windows 10 64-bit system. Software tools used included CUDA 11.0.194, CUDNN 8.0.5, Python 3.9.1, and Numpy 1.20.0. The experiments were implemented in the TensorFlow framework. Detection speed was measured with the same computer hardware.

2.3. Deep Learning Model

Mask R-CNN represents a state-of-the-art method in object detection, extending the Faster R-CNN framework by incorporating an additional branch at the model's terminal to achieve instance segmentation for each output proposal through fully connected (FC) layers. To enhance its suitability for real-time apple fruit segmentation, several adjustments and improvements were implemented in this work. A ResNet-DenseNet hybrid architecture was proposed as a replacement for the original backbone network to improve feature extraction. This modification enhances feature transferability and reusability while achieving superior performance with fewer parameters. Furthermore, since our objective focuses on apple recognition and segmentation in complex backgrounds—where the final segmentation output involves only a single class and does not require distinguishing between object categories—the classification branch within the multi-task framework was eliminated. A dedicated "apple" class was defined to streamline computational efficiency and minimize losses. The overall model architecture is depicted in Figure 4.

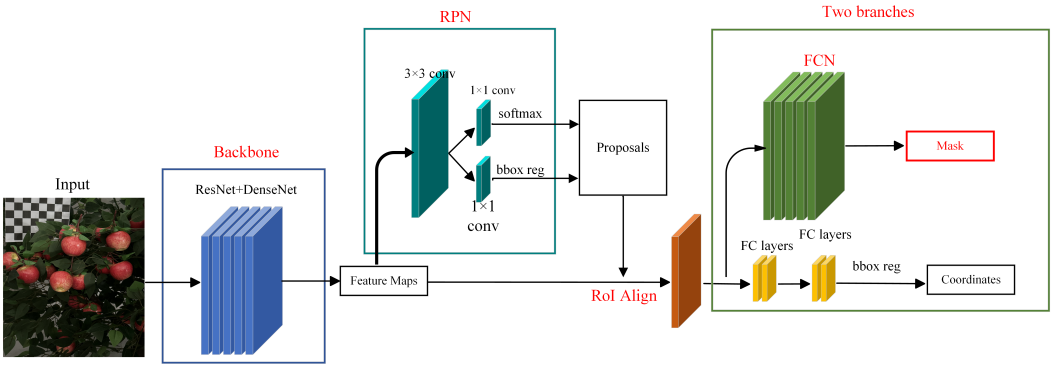


Figure 4. Improved Mask R-CNN network architecture.

2.3.1. Feature Extraction (ResNet + DenseNet)

Convolutional networks of varying depths can be constructed by designing distinct weight layers for image feature extraction. However, when conventional CNN networks exceed a certain depth, training error increases with additional convolutional layers, leading to reduced classification accuracy on test datasets. ResNet addresses this challenge by learning residual representations between inputs and outputs through multi-layer parameters, significantly enhancing training speed and prediction accuracy in deep networks. Its core innovation lies in establishing "shortcut connections" (skip connections) between layers, facilitating gradient backpropagation during training to enable deeper architectures. Nevertheless, during practical training, features from excessively small regions may be neglected due to resolution reduction during convolution. To better preserve small-region features, DenseNet—inspired by ResNet—is designed such that each layer directly receives outputs from all preceding layers. By connecting all layers in a feed-forward manner, DenseNet enhances feature reuse at all levels compared to ResNet with Feature Pyramid Networks (FPNs). This strengthens feature propagation and reuse, particularly beneficial for detecting occluded or undersized targets. Consequently, this study employs a backbone network integrating ResNet (to deepen training capability) and DenseNet (to retain low-

dimensional features) for feature extraction. Each Residual Dense Unit comprises multiple convolutional layers and ReLU activations. Its output establishes dense connections with every convolutional layer in subsequent units, enabling continuous information flow (as detailed in Figure 5).

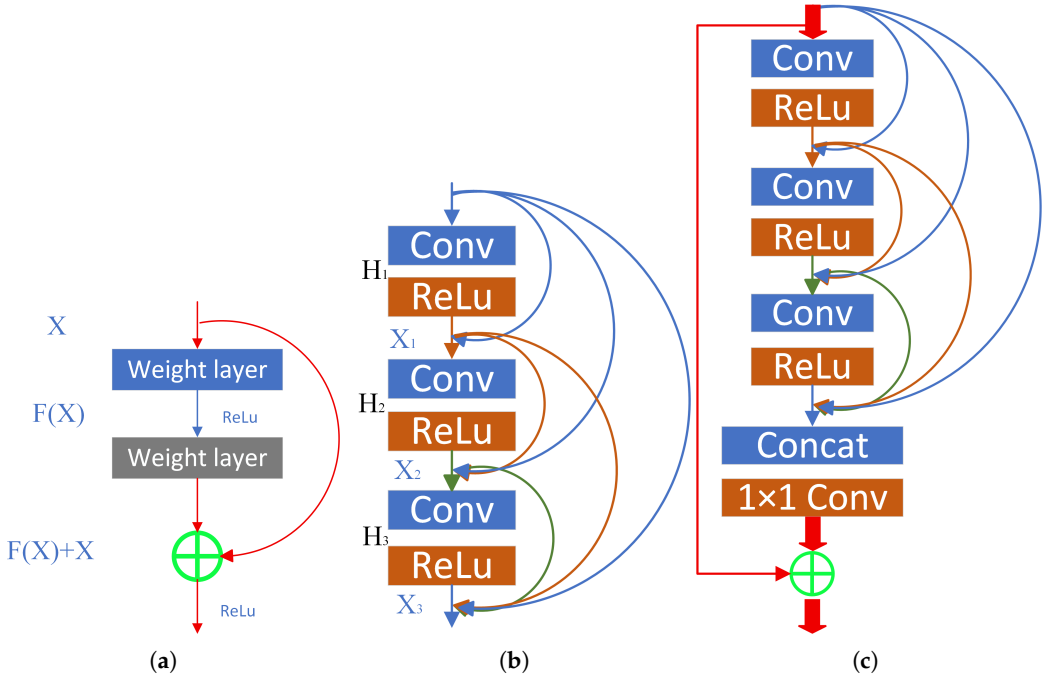


Figure 5. The specific network structure of backbone: (**a**) structure of residual block. (**b**) structure of dense block (l = 3). (**c**) structure of residual dense block.

For a convolutional network, let x_0 denote the input image. The network comprises L layers, with each layer implementing a nonlinear transformation $H(\cdot)$, where i represents the ith layer. The i layer receives feature maps $x_0, x_0, \cdots\cdots, x_{i-1}$ from all preceding layers as input, as

$$x_i = H_i([x_0, x_1, \cdots\cdots, x_{i-1}]) \quad (1)$$

where $[x_0, x_0, \cdots\cdots, x_{i-1}]$ represents the concatenation of these feature maps. Here, $H(\cdot)$ is defined as a composite function of two sequential operations: a ReLU activation followed by a 3 × 3 convolution (Conv).

Let F_{d-1} and F_d denote the input and output of the d-th dense unit, respectively, and G_0 the number of initial feature maps. The output of the c-th convolutional layer within the unit can be expressed as

$$F_{d,c} = H\{[F_{d-1}, F_{d,1}, \cdots, F_{d,c-1}]\} \quad (2)$$

Owing to the connectivity between the unit's input and its convolutional layers, feature maps require compression at the unit's terminus. Thus, the feature map count regulated by a 1 × 1 convolution is denoted as

$$F_{d,LF} = H^d_{LFF}\{[F_{d-1}, F_{d,1}, \cdots, F_{d,c}]\} \quad (3)$$

where H^d_{LFF} represents 1 × 1 convolution. The final output of the unit can be expressed as

$$F_d = F_{d-1} + F_{d,LF} \quad (4)$$

In this work, we employ a 3-layer dense block architecture, where each block consists of a sequence of BN + ReLU + Conv (3 × 3) operations. The input image resolution is set

to 512 × 512 pixels. To maximize information flow between layers, all network layers are directly interconnected. Maintaining feed-forward characteristics, each layer receives the concatenated feature maps from all preceding layers as input, while its own output feature maps serve as inputs to all subsequent layers. Following the dense blocks, a transition layer is incorporated, comprising a 1 × 1 convolutional kernel with 4k channels, where k denotes the growth rate. Setting k = 4 results in each dense layer produces feature maps with a dimensionality of 4.

2.3.2. Generation of RoIs and RoIAlign

Feature extraction of apple images is performed using a hybrid ResNet-DenseNet backbone, generating corresponding feature maps. The output from this backbone network serves as input to the Region Proposal Network (RPN), which produces candidate object regions with associated probability scores. To address scale variation across images (caused by varying shooting distances) and occlusion challenges (including vertical/horizontal overlapping structures), we employ three anchor scales (16 × 16, 64 × 64, 128 × 128) and three aspect ratios (1:1, 1:2, 2:1). Within the RPN, classification (CLS) and bounding-box regression (BBR) branches are randomly initialized to generate nine anchors per sliding-window position. These anchors correspond to a 2 × 9 probability matrix (objectness scores) and a 4 × 9 coordinate matrix (bounding-box vertices). After processing through the RPN (adding minimal computational overhead equivalent to a two-layer network), preliminary proposals are generated. These proposals are mapped onto the feature maps from the preceding stage, forming Regions of Interest (RoIs). The RoIs and their corresponding feature maps are fed into RoIAlign—a technique proposed to enhance pixel-accurate mask prediction. RoIAlign eliminates the harsh quantization of RoIPooling by precisely aligning extracted features with input coordinates through bilinear interpolation.

2.3.3. Target Detection and Instance Segmentation (FCN)

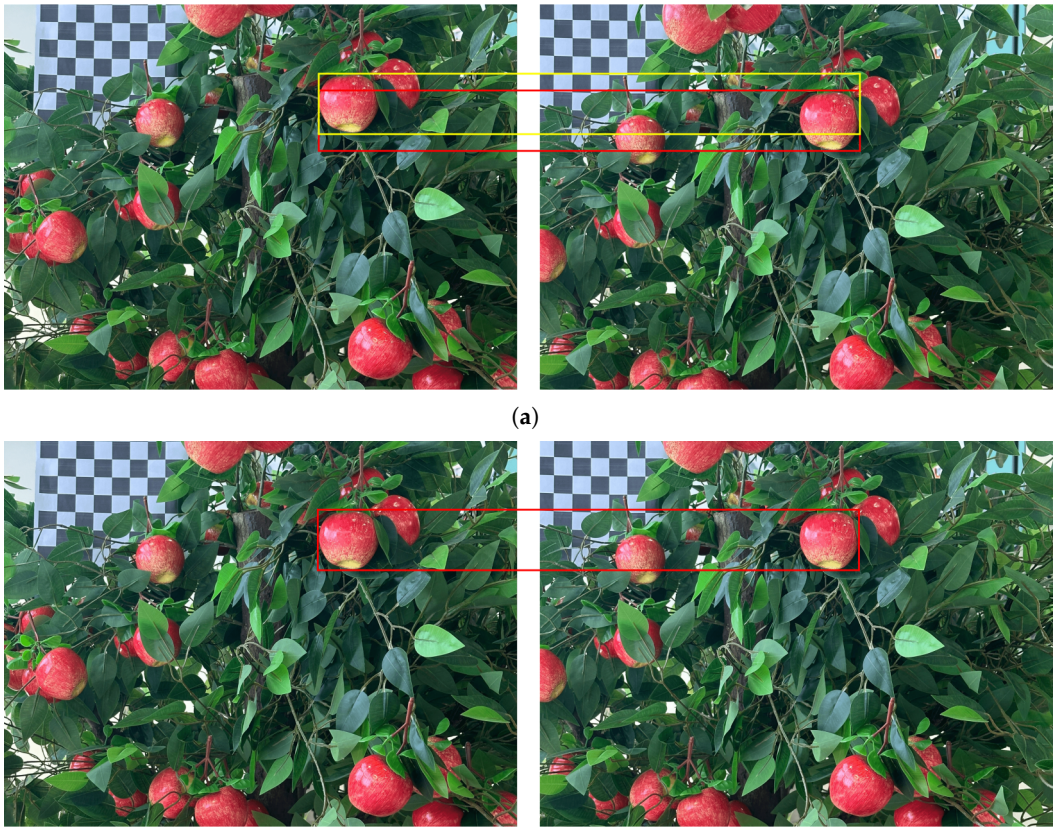
Following RoIAlign processing, instance segmentation is performed using a Fully Convolutional Network (FCN). While traditional Mask R-CNN employs three parallel branches for classification, bounding-box regression, and instance segmentation, this work streamlines the architecture to achieve single-target apple recognition in complex backgrounds. Specifically, the classification branch is eliminated to enhance computational efficiency without compromising segmentation accuracy. The adopted FCN provides end-to-end segmentation through an encoder-decoder structure: convolutional layers progressively downsample feature maps, followed by transposed convolutions (deconvolution) that upsample resolution via interpolation. Pixel-wise classification is ultimately applied to generate precise segmentation masks.

Segmentation is performed concurrently with recognition tasks. The Mask R-CNN framework comprises three stages . First, the input image is processed through the backbone network to extract multi-scale feature maps. Subsequently, the Region Proposal Network (RPN) generates candidate object proposals based on these feature maps, which are precisely aligned via the RoI Align layer (employing bilinear interpolation to avoid quantization errors). The aligned features are then fed into two parallel branches: a bounding box regression branch (refining proposal coordinates) and a mask branch (generating pixel-level binary masks using a Fully Convolutional Network, FCN). The final outputs consist of the target's localized bounding box coordinates and instance segmentation masks.

2.4. The Binocular Calibration and Binocular Positioning Principle

Binocular positioning first requires camera calibration to obtain rectified images, as stereoscopic images exhibit distortions (Figure 6a). This study employs Zhang's calibration method [48]. By capturing multiple images of a planar calibration board (e.g., a checker-

board), the intrinsic parameters (focal length, principal point, distortion coefficients) and extrinsic parameters (pose relationships between the camera and calibration board) are efficiently computed through a combination of closed-form solutions and nonlinear optimization. Subsequently, image distortions from the stereo camera are algorithmically corrected. After rectification, the positions of apples in the left and right images become horizontally aligned (Figure 6b).



(**a**)

(**b**)

Figure 6. Comparison of binocular image before and after correction: (**a**) before binocular image correction. (**b**) after binocular image correction.

The critical requirement for binocular vision lies in maintaining parallel optical axes and coplanar imaging planes between the two cameras. As illustrated in Figure 7, the binocular positioning principle employs left (O_L) and right (O_R) cameras arranged in a parallel configuration with identical orientation. The *baseline* distance between their optical centers and a shared focal length (f) constitute key system parameters. A spatial feature point $P(X_p, Y_p, Z_p)$ projects onto the left and right imaging planes as pixel coordinates $P_L(X_L, Y_L)$ and $P_R(X_R, Y_R)$, respectively, with their horizontal coordinate difference defining the $Disparity(X_L - X_R)$. Following triangulation principles, the three-dimensional coordinates of the feature point can be analytically determined through Equations (1)–(3).

$$Z_p = \frac{Baseline \times f}{Disparity} = \frac{Baseline \times f}{X_L - X_R} \tag{5}$$

$$X_p = \frac{Baseline}{Disparity} \times X_L = \frac{Baseline}{X_L - X_R} \times X_L \tag{6}$$

$$Y_p = \frac{Baseline}{Disparity} \times Y_L = \frac{Baseline}{X_L - X_R} \times Y_L \tag{7}$$

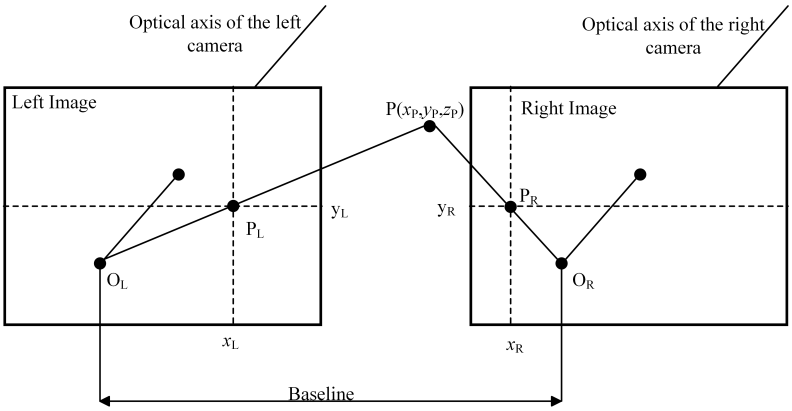


Figure 7. Binocular positioning principle.

It is evident that determining variables such as the focal length f, *baseline*, and *Disparity* is critically important for further computation of the 3D coordinates of P. Among these, the *baseline* and f can be directly obtained from the camera specifications. However, *Disparity* must be derived from the pixel coordinates of feature points in the left and right images, which are acquired through stereo matching.

2.5. Apple Stereo Matching

Stereo matching constitutes a critical technology in binocular vision systems. Its fundamental task lies in establishing projective correspondence between corresponding pixels in left and right views to resolve scene depth information through disparity calculation. Current mainstream matching algorithms include region-based Semi-Global Block Matching (SGBM), local window block matching, and feature descriptor-based strategies such as SIFT and SURF [49,50]. Addressing the morphological characteristics of apple targets in orchard environments, this study employs a template-matching approach to achieve cross-view association of fruit targets.The core principle of template matching involves sliding a template image over a target image to calculate similarity metrics between the template and each potential region in the target, thereby identifying the optimal match. During the sliding process, the matching cost at each position is computed, forming a matching cost matrix. The extremum positions within this matrix correspond to the matched apple locations.

The template-matching algorithm in the OpenCV vision library, implemented via the cv2.matchTemplate() function, provides six similarity metric-based matching criteria as follows: squared difference (TM_SQDIFF), normalized squared difference (TM_SQDIFF_NORMED), cross-correlation (TM_CCORR) and its normalized form (TM_CCORR_NORMED), as well as the correlation coefficient (TM_CCOEFF) and normalized correlation coefficient (TM_CCOEFF_NORMED). Through quantitative comparative analysis of response characteristics across different matching criteria in fruit imagery, this study found that the normalized squared difference method (TM_SQDIFF_NORMED) demonstrates superior template similarity discrimination under illumination-invariant conditions. Specifically, it achieves 12.6–25.3% higher matching accuracy compared to alternative approaches, proving particularly effective for target association tasks in orchard environments with complex lighting conditions, which was defined in Equation (4).

$$D(x,y) = \frac{\sum_{i,j}(T(i,j) - I(x+i,y+j))^2}{\sqrt{\sum_{i,j} T(i,j)^2 \times \sum_{i,j} I(x+i,y+j)^2}} \tag{8}$$

The similarity metric $D(x,y)$ for normalized squared difference matching is calculated as the normalized sum of squared differences between template pixels $T(i,j)$ and target region pixels $I(x+i,y+j)$, with its value range constrained to $D \in [0,1]$. Here, 0 indicates

perfect matching, while 1 denotes complete dissimilarity. When $D(x, y)$ persistently resides in the high-value domain, the system automatically activates an invalid match filtering mechanism to address occlusion-induced or cross-view absence of corresponding apple targets. To enhance matching efficiency, this study innovatively introduces a parallel epipolar constraint mechanism. Leveraging the prior knowledge in binocular geometry that the horizontal coordinate of a left-view feature point consistently exceeds its right-view counterpart ($X_L > X_R$), the template-matching search domain is restricted to ([0, X_2],[Y_1, Y_2]) (illustrated in Figure 8). This spatial constraint reduces computational complexity by approximately 62.3% compared to conventional full-image search strategies. By eliminating invalid search regions, the proposed method significantly improves algorithmic real-time performance while maintaining matching accuracy.



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

Figure 8. Template matching with parallel epipolar constraints: (**a**) The left picture shows the template apple. (**b**) The corrected binocular image, which conforms to the epipolar constraint (the apples in the binocular image are at the same height).

The instance segmentation results generated by the Mask R-CNN framework exhibit dual representations: geometric positioning information via bounding boxes (BBoxes) and pixel-level contour descriptions through masks. As visualized in Figure 9, each apple instance's BBox is defined by its top-left coordinate (X_{min},Y_{min}) and bottom-right coordinate (X_{max},Y_{max}), maintaining strict instance-level topological consistency with the corresponding mask matrix.

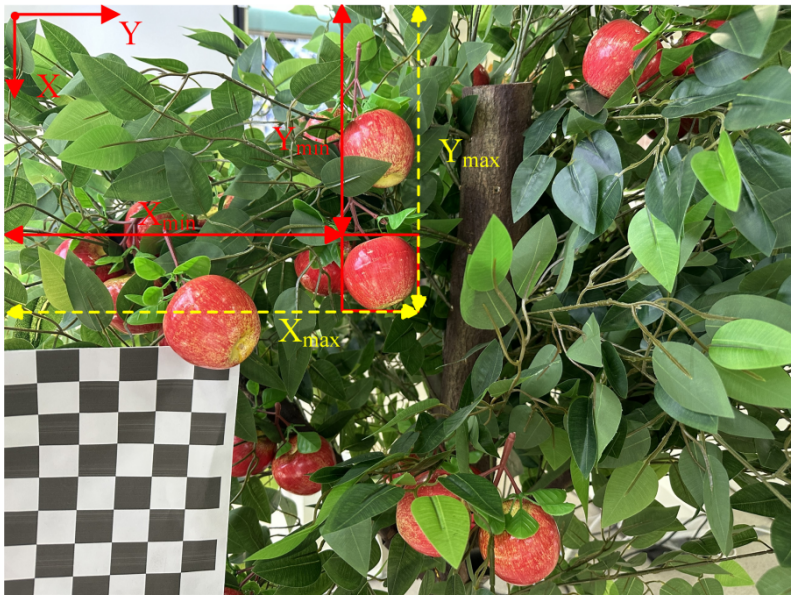


Figure 9. The visualization of values in the BBox.

The template-matching process selects the bounding box (BBox) generated by Mask R-CNN in the left image as the template to search for similar regions in the right image, aiming to identify the most-matching region (i.e., the matching BBox) and thereby establish correspondence between apples in the left and right views. However, since the BBoxes and masks in the right image are independently generated by Mask R-CNN, discrepancies may exist between the matching BBox derived from template matching and the right-image BBox. Consequently, the left BBox and its corresponding mask can only align with the template-matching result (the matched BBox), necessitating a secondary matching operation between the matched BBox and the right-image BBox to achieve their association. Notably, during this secondary matching phase, three types of invalid apple data should be eliminated to ensure data validity:

1. Apples present only in the left image: Apples detected by Mask R-CNN in the left image (with generated BBoxes and masks) lack corresponding regions in the right image due to template-matching failures (excessive matching cost or absence of similar regions within the search range). This occurs when apples are occluded in the right image or lie outside its field of view. Such apples introduce depth calculation errors, resulting in invalid 3D coordinates.
2. Apples present only in the right image: Apples detected in the right image (with generated BBoxes and masks) lack corresponding templates in the left image (i.e., undetected by Mask R-CNN in the left image). This arises from occlusion in the left image or template-matching failures. Without template support from the left image, these apples cannot establish correspondence through template matching, thereby introducing spurious targets.
3. Matched apples undetected in the right image (Mask R-CNN false negatives): While template matching identifies regions in the right image resembling the left template (matching BBox), Mask R-CNN fails to generate corresponding BBoxes and masks. Causes include missed detections by Mask R-CNN in the right image or erroneous classification of background regions as apples by template matching. These apples yield entirely erroneous positioning coordinates.

The secondary matching method involves iterating through all BBoxes in the right image, calculating their Intersection over Union (*Iou*) with the matching BBox, and selecting the right BBox with the maximum *Iou* as the final matching result. Upon completing secondary matching, the correspondence between BBoxes and masks in the left and right images is definitively established.

2.6. Apple Positioning

After completing stereo matching of apples, the depth information can be derived from the disparity data obtained through stereo matching using binocular positioning principles. The depth map visually represents the spatial distribution of apples and their surrounding environment, providing enhanced visual reference for subsequent apple positioning. The depth information is illustrated in Figure 10b.

To improve positioning accuracy and reduce errors, four feature points of each apple are selected, and their 3D coordinates are calculated, with the final positioning result determined by averaging these coordinates. The methodology for feature point selection is demonstrated in Figure 10a. Within the bounding box (BBox) of an apple, a vertical traversal is performed to identify the feature points with the minimum and maximum y-coordinates, while a horizontal traversal locates the points with the minimum and maximum x-coordinates. These four points represent the apple's extreme positions (leftmost, rightmost, topmost, and bottommost), effectively encompassing its geometric profile and mitigating positioning deviations caused by partial occlusion or irregular shapes.



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

Figure 10. Depth information of binocular images: (**a**) Original image of apple (feature point selection). (**b**) Apple image depth information.

2.7. Evaluation Criteria of Detection Model

This study evaluates the detection and segmentation performance of improved Mask R-CNN using Intersection over Union (IoU) and Average Precision (AP), as formalized in Equations (5) and (6), respectively.

$$IoU = \frac{I}{U} = \frac{Output \cap GT}{Output \cup GT} \tag{9}$$

$$AP = \int_1^0 P(R)dR \tag{10}$$

Here, $Output$ refers to the masks or bounding boxes generated by improved Mask R-CNN, while GT (ground truth) represents the manually annotated polygonal and rectangular labels for apples in the dataset. AP (average precision) denotes the average precision for object detection or segmentation. Precision (P) and recall (R) are defined in Equations (7) and (8), respectively.

$$P = \frac{TP}{TP + FP} \tag{11}$$

$$R = \frac{TP}{TP + FN} \tag{12}$$

The terms true positives (TP), false positives (FP), and false negatives (FN) are derived by comparing improved Mask R-CNN outputs with ground truth data, representing correctly detected, erroneously detected, and undetected objects, respectively. An IoU threshold of 0.5 was applied for calculating AP, P, and R.

2.8. Evaluation Criteria of Apple Binocular Positioning

This study employs the coefficient of variation (CoV) and positioning accuracy (PA) to evaluate positioning performance. As a dimensionless metric, the CoV normalizes the standard deviation to the mean, enabling equitable comparison of positioning precision across different apples or datasets. A smaller CoV indicates that the depth values of the four feature points are closely clustered, reflecting stable positioning results, while a larger CoV suggests significant discrepancies in depth values, which may lead to unreliable positioning. The mathematical definitions of CoV and PA are provided in Equations (10) and (11), respectively.

$$COV = \frac{S}{\overline{P_Z}} \tag{13}$$

$$PA = \frac{1 - max(P_{Z_{max}} - \overline{P_Z}, P_{Z_{min}} - \overline{P_Z})}{\overline{P_Z}} \tag{14}$$

Here, S represents the standard deviation of the four feature points for each apple, while P_Z denotes the average depth value of these four points. $P_{Z_{max}}$ and $P_{Z_{min}}$ correspond to the maximum and minimum depth values among the four feature points of each apple. The operator $max()$ returns the maximum value among the comma-separated entries within the parentheses.

2.9. Picking Experiment

This study conducted apple positioning and picking experiments using a self-designed apple-picking robot, as illustrated in Figure 11. The apple-picking robotic system developed in this study employs a modular architecture comprising three core functional modules: a mobile chassis, a 6-degree-of-freedom (6-DoF) manipulator, and an end-effector. The crawler-type mobile chassis integrates critical subsystems, including a power supply unit, manipulator control module, and central control system. A custom-designed 6-DoF manipulator is mounted on this chassis, with the end-effector attached to its terminal link to form an integrated picking unit.



Figure 11. Self-designed apple-picking robot.

The mobile base features a crawler-type chassis composed of a chassis compartment and a crawler locomotion mechanism. The chassis compartment houses environmental sensing systems and motion control units for the robot. The crawler locomotion mechanism integrates load-bearing wheels, drive wheels, tension auxiliary wheels, and belt support wheels. The manipulator incorporates six rotational joints, corresponding to 6 DoF, including the base (Joint 1), shoulder (Joint 2), elbow (Joint 3), wrist 1 (Joint 4), wrist 2 (Joint 5), and wrist 3 (Joint 6). The robotic arm control process first involves path planning within the joint space for the picking manipulator, achieved using the APF-A* algorithm. Subsequently, the required pose information for the robotic arm is transmitted via serial communication to the individual joint actuators. These actuators drive the corresponding joints through the calculated angles, thereby accomplishing the apple-picking task.

The end-effector incorporates a bionic flexible two-finger gripper. Mimicking the joint distribution of a human finger, the gripper base achieves 360° rotation via a brushless DC motor to attain suitable grasping postures. The symmetrical two-finger design adaptively envelops the fruit, increasing the contact area by 40%. The gripper's opening/closing stroke ranges from 0 to 120 mm, enabling adaptation to apples of varying sizes. Furthermore, the full-stroke opening/closing response time is within 0.5 s, meeting the requirement for rapid

apple grasping. Grip force control employs a fuzzy PID adjustment strategy. Real-time adjustments to the gripping force are made based on the magnitude of the pressure error. Concurrently, PID parameters are dynamically updated based on fruit firmness detected by the pressure sensor, effectively maintaining overshoot at a low level.

To validate the effectiveness of the proposed localization method, this approach was implemented on an apple-picking robot for picking experiments. The robot performed individual apple-target picking across 70 experimental trials. To evaluate the localization performance under diverse conditions, these 70 picking operations were further categorized according to varying occlusion scenarios. According to the proportion of fruit occlusion area, it is divided into slight occlusion (occlusion area < 30%), moderate occlusion (30–60%), severe occlusion (60–90%), and complete occlusion (above 90%). Across the 70 picking trials, the occlusion conditions were classified as follows: 10 trials under light occlusion, 20 trials under moderate occlusion, 35 trials under severe occlusion, 5 trials under full occlusion. During picking experiments, the robot's host computer first processes images of apples captured by binocular cameras. An improved Mask R-CNN model detects the target apples within the images. Subsequently, the 3D coordinates of the apples are calculated based on binocular vision. The host computer then determines the deviation between the target fruit's coordinates and the image center. This deviation is converted into corresponding joint motion angles for the robotic arm using a specified scaling factor. Path planning for the robotic arm within the joint space is achieved using the APF-A* algorithm. The required pose information for the robotic arm is transmitted via serial communication to the individual joint actuators. These actuators drive the corresponding joints to move by the calculated angles, enabling the robotic arm to approach the target along the planned trajectory. Finally, the claw-type end-effector completes damage-free grasping of the fruit, assisted by the pressure sensor.

3. Results and Discussion

3.1. Training Assessment and Performance of Detection Model

The loss function serves as a metric to evaluate the effectiveness of model predictions on a dataset, where a lower loss value indicates closer alignment between model outputs and ground truth values. As shown in Figure 12, the training loss curve of the improved Mask R-CNN has converged. The horizontal axis represents the number of iterations (each consisting of 100 steps), and the loss value progressively decreases as iterations increase. After approximately 400 iterations, the loss stabilizes around a minimum value of 0.45.

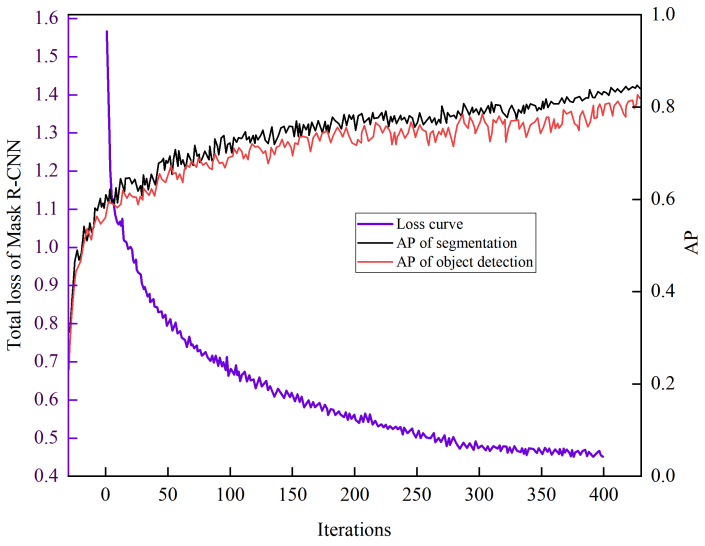


Figure 12. Loss curve and AP curve of the improved Mask R-CNN in every iteration during training.

The converged loss curve demonstrates that the improved Mask R-CNN effectively identifies targets and performs apple image detection. The improved model achieves Average Precision (*AP*) scores of 84.46% and 81.96% for object detection and instance segmentation, respectively. To validate performance, a set of binocular apple images randomly selected from the validation dataset was tested. The model's testing results are illustrated in Figure 13.

Furthermore, the Mask R-CNN was evaluated on the test dataset, achieving detection and segmentation Intersection over Union (*IoU*) scores of 86.17% and 82.75%, respectively, in this study.

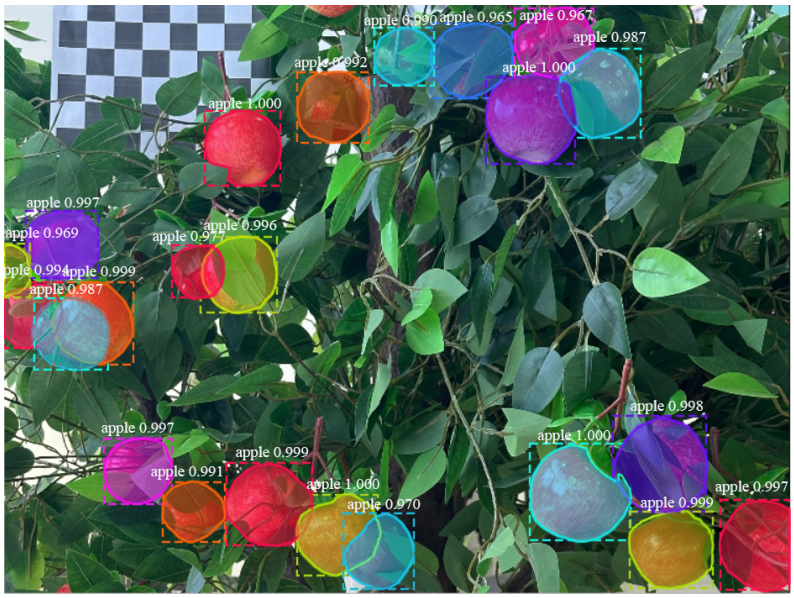


Figure 13. Instance segmentation result of the binocular apple image by the improved Mask R-CNN.

3.2. Performance Evaluation of Binocular Positioning

This study visually presents the depth information of four characteristic points of apples within images and evaluates the positioning performance. Following detection, segmentation, stereo matching, and positioning processes, the final visualization of daytime and nighttime positioning results is shown in Figure 14. Red lines connect the top and bottom characteristic points of apples, while green lines link left-right characteristic points. In the left subfigure of Figure 14, apples marked with green lines indicate unmatched counterparts in the right subfigure. Additionally, some unmarked apples in the right image are discarded during binocular positioning due to absence in the left image. To quantify positioning performance, two metrics are introduced: the Coefficient of Variation (*CoV*) and Positioning Accuracy (*PA*). The *CoV* advantageously provides dimensionless measurement of relative dispersion among depth values from multiple characteristic points of the same target, thereby complementing *PA*'s limitations and enabling comprehensive quantitative assessment. Specifically, higher *CoV* values coupled with lower *PA* values indicate significant discrepancies in multi-point positioning results for individual apples. Conversely, lower *CoV* values paired with higher *PA* values demonstrate superior consistency across characteristic point positioning, reflecting optimized positioning performance.

This paper statistically analyzed the *CoV* and *PA* for 100 image samples from the dataset, as shown in Figure 15. All datasets used for *CoV* and *PA* calculations excluded the training data. The averaged *CoV* and *PA* across the 100 samples were 5.09% and 99.61%, respectively. These results demonstrate that the positioning performance largely meets the requirements of apple-picking robotic systems.

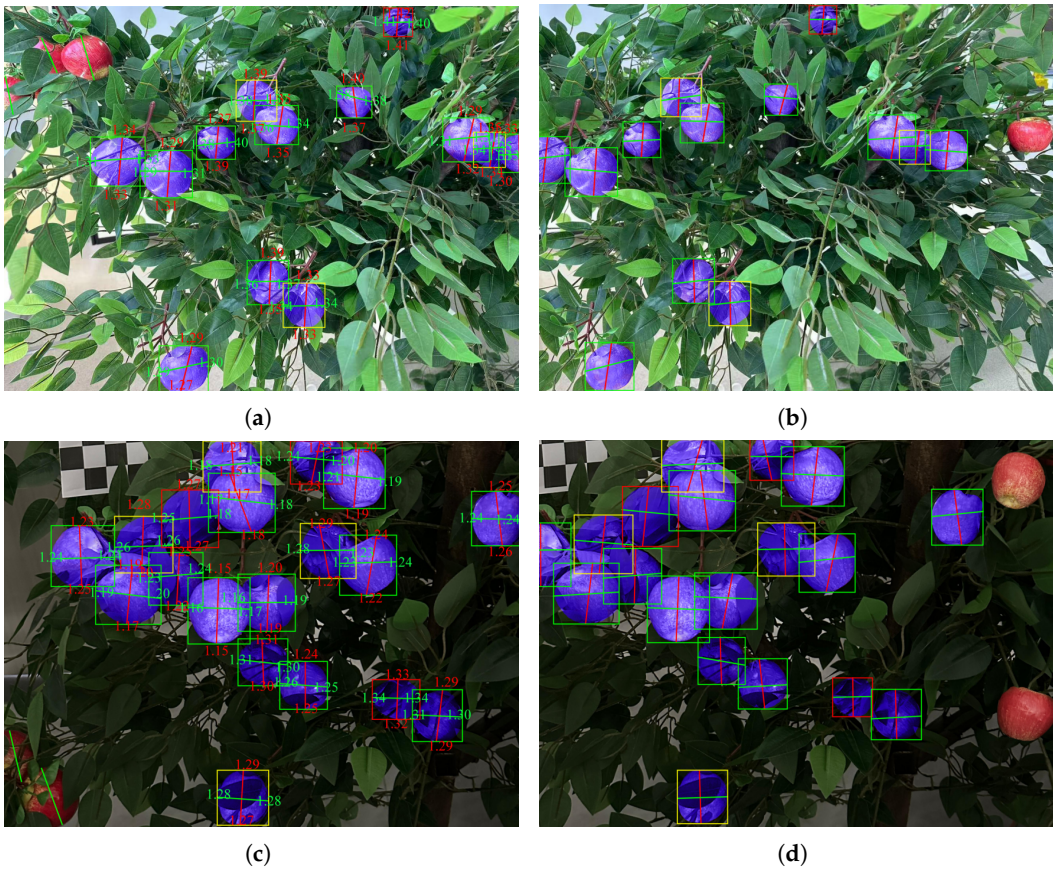


Figure 14. Visualization of binocular positioning: (**a**) Left image (daytime). (**b**) Right image (daytime). (**c**) Left image (night). (**d**) Right image (night).

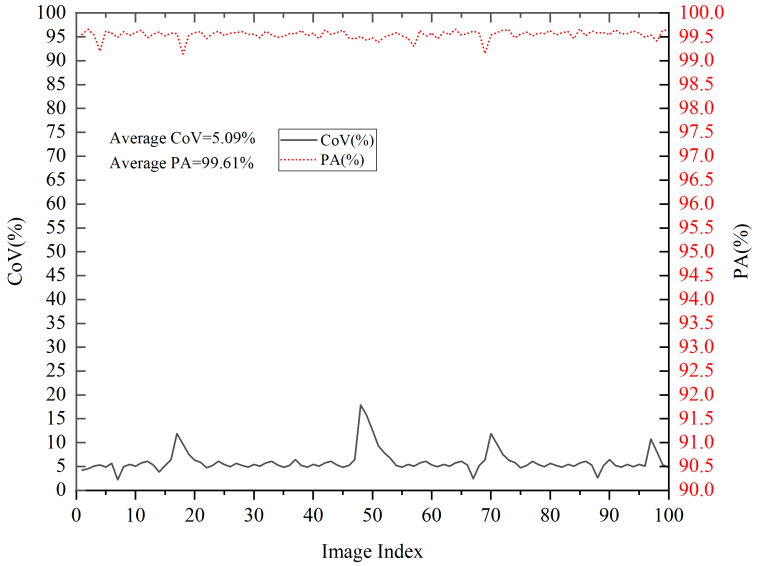


Figure 15. *CoV* and *PA* of the 100 images of the test dataset.

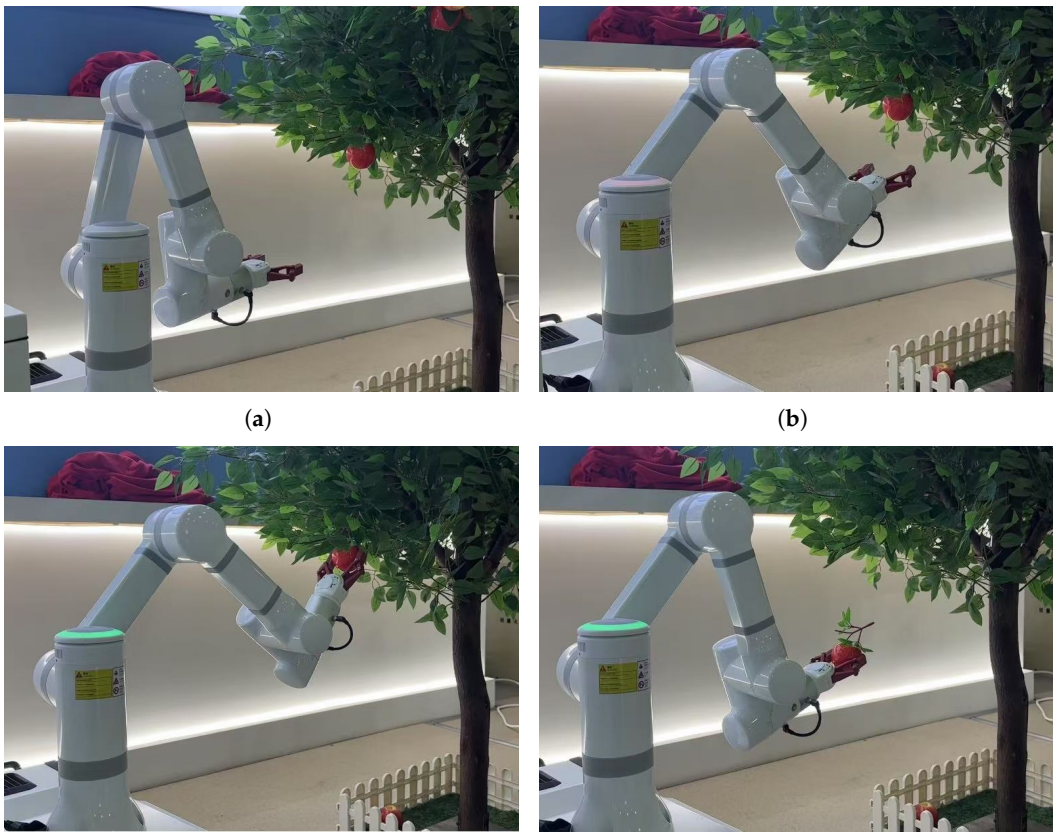
3.3. Robot Automatic Picking Experiment

This experiment utilized artificial tree models to construct a simulated orchard environment, where the robot performed picking trials on individual apples under varying occlusion conditions. Among the 70 picking trials, 66 were successfully completed while 4 resulted in picking failure. Positioning failures occurred only under extreme occlusion scenarios, where excessive positioning errors resulted in picking failure, with the average

single-image processing time measured at 0.36 s and the mean picking cycle time per apple recorded at 7.7 s. The overall picking success rate reached 94.3%, confirming that the proposed localization method satisfies real-time operational requirements for apple-picking robotic systems. Table 1 shows some of the picking experiment data. Figure 16 demonstrates the full workflow of the robotic picking process.

Table 1. Part of the picking experiment data.

Number	Degree of Fruit Occlusion	Image Processing Time (s)	Total Time Cost (s)
1	slight occlusion	0.34	7.73
2	moderate occlusion	0.37	7.91
3	moderate occlusion	0.36	7.75
4	moderate occlusion	0.36	7.88
5	severe occlusion	0.42	8.11



(**a**) (**b**) (**c**) (**d**)

Figure 16. Picking experiment of apple-picking robots: (**a**) Starting position. (**b**) Looking for a target. (**c**) Detection process. (**d**) Successful capture.

3.4. Results from Other Studies

Compared to conventional image processing technology-driven binocular positioning strategies, visual positioning systems based on deep learning demonstrate superior system performance owing to their exceptional recognition accuracy. However, research on deep learning-based binocular positioning for fruit-picking robots remains insufficient. Li et al. developed an apple 3D coordinate measurement system by integrating the Faster R-CNN object detection framework with traditional image segmentation algorithms, with experimental results showing a mean depth measurement standard deviation of 0.51 cm

and spatial positioning accuracy reaching 99.64% [51]. It should be noted that traditional segmentation algorithms exhibit technical limitations in scenarios involving sudden illumination changes and target occlusion, which may directly compromise the environmental adaptability of positioning systems and the reproducibility of measurement outcomes. Yu et al. designed a binocular positioning method for apple-picking robots using traditional image processing techniques, with positioning errors below 7 mm in the X/Y axes and 5 mm in the Z-axis [22]. Zhang et al. developed an improved YOLOv5 model (LE-YOLO) integrated with binocular cameras to enable apple detection and positioning under suboptimal lighting conditions, reporting mean positioning errors of 0.32 cm and 0.65 cm under normal and adverse illumination, respectively [52]. Hu et al. introduced an improved YOLOX-based method with RGB-D imaging for apple detection and positioning, achieving depth errors below 5 mm [53]. Zhang et al. proposed a segmentation-based deep learning method for binocular precise positioning of apples, achieving a positioning accuracy of 99.49% [54]. However, the method exhibits prolonged computational time and insufficient adaptability to complex lighting conditions. While these methods demonstrate commendable performance in apple binocular positioning, they lack sufficient robustness in scenarios involving complex illumination or occlusion.

This study introduces improvements to the binocular positioning method for apple-picking robots, achieving a positioning accuracy of 99.61%. Leveraging a comprehensive dataset, the proposed approach adapts to diverse lighting conditions, fruit sizes, and shapes to detect and segment apples, followed by binocular vision-based positioning. The improved Mask R-CNN effectively segments occluded apple contours, with its streamlined processing pipeline discarding irrelevant background interference. This enables robust binocular apple positioning across various challenging environments. Technical specifications of these methods are systematically compared in Table 2.

Table 2. Results from other studies on fruit positioning.

Reference	Camera	Model/Method	Criteria
Li et al. [51]	Binocular camera	Faster R-CNN	Precision: 99.64%
Yu et al. [22]	Binocular camera	Traditional image processing	Depth error: less than 7 mm
Zhang et al. [52]	Binocular camera	LE-YOLO	Depth error: 0.65 cm
Hu et al. [53]	RGB-D camera	YOLOX	Depth error: less than 5 mm
Zhang et al. [54]	Binocular camera	Mask R-CNN	PA: 99.49%
Our method	Binocular camera	Improved Mask R-CNN	PA: 99.61%

Furthermore, in complex orchard environments, the appearance of apples may vary due to changes in illumination conditions, viewing angles, and occlusions. Reliance on a single sensor could potentially lead to system failure under certain circumstances. Future research could consider incorporating multimodal data fusion to further enhance positioning accuracy and robustness. This approach can effectively integrate texture and color information provided by different sensors, enabling more comprehensive characterization of apple features, thereby improving detection and positioning precision. For instance, while RGB images offer rich color information, near-infrared (NIR) images can provide complementary textural details. The synergistic combination of these complementary data sources may facilitate more accurate apple identification.

4. Conclusions

In this study, a DenseNet hybrid architecture was adopted to substitute the original backbone network, and the classification branch within the multi-task framework was eliminated to enhance the suitability of Mask R-CNN for real-time apple fruit segmentation. The improved Mask R-CNN was implemented on an apple-picking robot to detect and segment apples. A template-matching method based on parallel epipolar constraints was subsequently applied to perform preliminary matching of apples in binocular images. After eliminating three categories of invalid apple data, a secondary matching process integrated the template-matching results with model detection outputs, thereby establishing correlations between the bounding boxes (BBoxes) and masks in the left and right images. Finally, four representative feature points of the apples were selected, and their 3D coordinates were computed. The localization performance was evaluated on a dataset of 100 samples using the average Coefficient of Variation (CoV) and average Positioning Accuracy (PA). Finally, to validate the effectiveness of this localization method, it was applied to an apple-picking robot. Seventy picking trials were designed under four distinct occlusion scenarios. Experimental results demonstrate that the integration of the improved Mask R-CNN with binocular vision significantly enhanced apple positioning accuracy while simultaneously meeting the real-time operational requirements of the apple-picking robot. Future research will prioritize the integration of multimodal data fusion into this framework to further refine positioning precision and robustness. This research will enable development of a high-precision and cost-effective design methodology for vision-based positioning systems in apple-picking robots.

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

Funding: This research was funded in part by Postgraduate Research and Practice Innovation Program of Jiangsu Province (SJCX25_2516), and in part by the Jiangsu Provincial Science and technology project modern agriculture (BE2020406).

Data Availability Statement: The datasets presented in this article are not readily available because the data are part of an ongoing study. Requests to access the datasets should be directed to 231210301205@stu.just.edu.cn.

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

References

1. Kok, E.; Chen, C. Occluded apples orientation estimator based on deep learning model for robotic harvesting. *Comput. Electron. Agric.* **2024**, *219*, 108781. [CrossRef]
2. Lu, S.; Gao, P.; Liu, G.; Han, Z. The relationship between inputs and benefits of apple orchard and the countermeasures and suggestions for cost saving and efficiency increasing. *China Fruits* **2024**, *4*, 1–6.
3. UN Food & Agriculture Organization. The Impact of Automation on the Apple Industry 2022. Available online: https://openknowledge.fao.org/home (accessed on 12 February 2025).
4. Zhang, Z.; Igathinathane, C.; Li, J.; Cen, H.; Lu, Y.; Flores, P. Technology progress in mechanical harvest of fresh market apples. *Comput. Electron. Agric.* **2020**, *175*, 105606. [CrossRef]
5. Kang, H.; Chen, C. Fast implementation of real-time fruit detection in apple orchards using deep learning. *Comput. Electron. Agric.* **2020**, *168*, 105108. [CrossRef]
6. Open X-Embodiment: Robotic Learning Datasets and RT-X Models. Available online: https://robotics-transformer-x.github.io (accessed on 18 June 2025).
7. Physical Intelligence. Available online: https://www.physicalintelligence.company (accessed on 18 June 2025).

8. Zhang, J.; Wan, W.; Tanaka, N.; Fujita, M.; Takahashi, K.; Harada, K. Integrating a pipette into a robot manipulator with uncalibrated vision and TCP for liquid handling. *IEEE Trans. Autom. Sci. Eng.* **2023**, *21*, 5503–5522. [CrossRef]
9. Choi, D.W.; Park, J.H.; Yoo, J.H.; Ko, K. AI-driven adaptive grasping and precise detaching robot for efficient citrus harvesting. *Comput. Electron. Agric.* **2025**, *232*, 110131. [CrossRef]
10. Zhao, D.; Lv, J.; Ji, W.; Zhang, Y.; Chen, Y. Design and control of an apple harvesting robot. *Biosyst. Eng.* **2011**, *110*, 112–122.
11. Abeyrathna, R.M.R.D.; Nakaguchi, V.M.; Liu, Z.; Sampurno, R.M.; Ahamed, T. 3D Camera and Single-Point Laser Sensor Integration for Apple Localization in Spindle-Type Orchard Systems. *Sensors* **2024**, *24*, 3753. [CrossRef]
12. Zhang, K.; Lammers, K.; Chu, P.; Li, Z.; Lu, R. System design and control of an apple harvesting robot. *Mechatronics* **2021**, *79*, 102644. [CrossRef]
13. Gené-Mola, J.; Gregorio, E.; Guevara, J.; Auat, F.; Sanz-Cortiella, R.; Escolà, A.; Llorens, J.; Morros, J.R.; Ruiz-Hidalgo, J.; Vilaplana, V.; et al. Fruit detection in an apple orchard using a mobile terrestrial laser scanner. *Biosyst. Eng.* **2019**, *187*, 171–184. [CrossRef]
14. Yuan, Y.; Liu, H.; Yang, Z.; Zheng, J.; Li, J.; Zhao, L. A detection method for occluded and overlapped apples under close-range targets. *Pattern Anal. Appl.* **2024**, *27*, 12. [CrossRef]
15. Jiang, D.; Zheng, Z.; Li, G.; Sun, Y.; Kong, J.; Jiang, G.; Xiong, H.; Tao, B.; Xu, S.; Yu, H.; et al. Gesture recognition based on binocular vision. *Cluster Comput.* **2019**, *22*, 13261–13271. [CrossRef]
16. Tang, Y.; Chen, M.; Wang, C.; Luo, L.; Li, J.; Lian, G.; Zou, X. Recognition and localization methods for vision-based fruit picking robots: A review. *Front. Plant Sci.* **2020**, *11*, 510. [CrossRef]
17. Wang, F.; Chen, X.; Tan, C.; Li, J.; Zhang, Y. Hexagon-shaped screw recognition and positioning system based on binocular vision. In Proceedings of the 2018 37th Chinese Control Conference (CCC), Wuhan, China, 25–27 July 2018; pp. 5481–5486.
18. Takahashi, T.; Zhang, S.; Fukuchi, H. Measurement of 3-D locations of fruit by binocular stereo vision for apple harvesting in an orchard. In Proceedings of the 2002 ASA Annual Meeting, Chicago, IL, USA, 16–19 August 2022; American Society of Agricultural and Biological Engineers: Saint Joseph, MI, USA, 2002.
19. Williams, H.; Ting, C.; Nejati, M.; Jones, M.H.; Penhall, N.; Lim, J.; Seabright, M.; Bell, J.; Ahn, H.S.; Scarfe, A.; et al. Improvements to and large-scale evaluation of a robotic kiwifruit harvester. *J. Field Rob.* **2020**, *37*, 187–201. [CrossRef]
20. Lei, X.; Wu, M.; Li, Y.; Liu, A.; Tang, Z.; Chen, S. Detection and Positioning of *Camellia oleifera* Fruit Based on LBP Image Texture Matching and Binocular Stereo Vision. *Agronomy* **2023**, *13*, 2153. [CrossRef]
21. Luo, L.; Tang, Y.; Zou, X.; Ye, M.; Feng, W. Vision-based extraction of spatial information in grape clusters for harvesting robots. *Biosyst. Eng.* **2016**, *151*, 90–104. [CrossRef]
22. Yu, X.; Fan, Z.; Wang, X.; Wan, H.; Wang, P.; Zeng, X.; Jia, F. A lab-customized autonomous humanoid apple harvesting robot. *Comput. Electr. Eng.* **2021**, *96*, 107459. [CrossRef]
23. Zhao, G.; Yang, R.; Jing, X.; Zhang, H.; Wu, Z.; Sun, X.; Jiang, H.; Li, R.; Wei, X.; Fountas, S.; et al. Phenotying of individual apple tree in modern orchard with novel smartphone-based heterogeneous binocular vision and YOLOv5s. *Comput. Electron. Agric.* **2023**, *209*, 107814. [CrossRef]
24. Henila, M.; Chithra, P. Segmentation using fuzzy cluster-based thresholding method for apple fruit sorting. *IET Image Proc.* **2020**, *14*, 4178–4187. [CrossRef]
25. Guerra Ibarra, J.P.; Cuevas de la Rosa, F.J.; Linares Ramirez, A. Color Dominance-Based Polynomial Optimization Segmentation for Identifying Tomato Leaves and Fruits. *Agriculture* **2024**, *14*, 1911. [CrossRef]
26. Fu, L.; Tola, E.; Al-Mallahi, A.; Li, R.; Cui, Y. A novel image processing algorithm to separate linearly clustered kiwifruits. *Biosyst. Eng.* **2019**, *183*, 184–195. [CrossRef]
27. Mizushima, A.; Lu, R. An image segmentation method for apple sorting and grading using support vector machine and Otsu's method. *Comput. Electron. Agric.* **2013**, *94*, 29–37. [CrossRef]
28. Chapelle, O.; Haffner, P.; Vapnik, V.N. Support vector machines for histogram-based image classification. *IEEE Trans. Neural Netw.* **1999**, *10*, 1055–1064. [CrossRef]
29. Otsu, N. A threshold selection method from gray-level histograms. *Automatica* **1975**, *11*, 23–27. [CrossRef]
30. Koirala, A.; Walsh, K.B.; Wang, Z.; McCarthy, C. Deep learning–Method overview and review of use for fruit detection and yield estimation. *Comput. Electron. Agric.* **2019**, *162*, 219–234. [CrossRef]
31. Lv, J.; Xu, H.; Xu, L.; Zou, L.; Rong, H.; Yang, B.; Niu, L.; Ma, Z. Recognition of fruits and vegetables with similar-color background in natural environment: A survey. *J. Field Rob.* **2022**, *39*, 888–904. [CrossRef]
32. Gheisari, M.; Wang, G.; Bhuiyan, M.Z.A. A Survey on Deep Learning in Big Data. In Proceedings of the 2017 IEEE International Conference on Computational Science and Engineering (CSE) and IEEE International Conference on Embedded and Ubiquitous Computing, (EUC), Guangzhou, China, 21–24 July 2017; Volume 2, pp. 173–180.
33. Garcia-Garcia, A.; Orts-Escolano, S.; Oprea, S.; Villena-Martinez, V.; Garcia-Rodriguez, J. A review on deep learning techniques applied to semantic segmentation. *arXiv* **2017**, arXiv:1704.06857.
34. Suo, R.; Fu, L.; He, L.; Li, G.; Majeed, Y.; Liu, X.; Zhao, G.; Yang, R.; Li, R. A novel labeling strategy to improve apple seedling segmentation using BlendMask for online grading. *Comput. Electron. Agric.* **2022**, *201*, 107333. [CrossRef]

35. Gao, F.; Fu, L.; Zhang, X.; Majeed, Y.; Li, R.; Karkee, M.; Zhang, Q. Multi-class fruit-on-plant detection for apple in SNAP system using Faster R-CNN. *Comput. Electron. Agric.* **2020**, *176*, 105634. [CrossRef]
36. Sun, X.; Fang, W.; Gao, C.; Fu, L.; Majeed, Y.; Liu, X. Remote estimation of grafted apple tree trunk diameter in modern orchard with RGB and point cloud based on SOLOv2. *Comput. Electron. Agric.* **2022**, *199*, 107209. [CrossRef]
37. Sun, K.; Wang, X.; Liu, S.; Liu, C. Apple, peach, and pear flower detection using semantic segmentation network and shape constraint level set. *Comput. Electron. Agric.* **2021**, *185*, 106150. [CrossRef]
38. Chen, Z.; Ting, D.; Newbury, R.; Chen, C. Semantic segmentation for partially occluded apple trees based on deep learning. *Comput. Electron. Agric.* **2021**, *181*, 105952. [CrossRef]
39. Dias, P.A.; Tabb, A.; Medeiros, H. Multispecies fruit flower detection using a refined semantic segmentation network. *IEEE Rob. Autom. Lett.* **2018**, *3*, 3003–3010. [CrossRef]
40. Kang, H.; Chen, C. Fruit detection, segmentation and 3D visualisation of environments in apple orchards. *Comput. Electron. Agric.* **2020**, *171*, 105302. [CrossRef]
41. Gené-Mola, J.; Sanz-Cortiella, R.; Rosell-Polo, J.R.; Morros, J.R.; Ruiz-Hidalgo, J.; Vilaplana, V.; Gregorio, E. Fruit detection and 3D location using instance segmentation neural networks and structure-from-motion photogrammetry. *Comput. Electron. Agric.* **2020**, *169*, 105165. [CrossRef]
42. Wang, J.; Wang, L.; Han, Y.; Zhang, Y.; Zhou, R. On combining deepsnake and global saliency for detection of orchard apples. *Appl. Sci.* **2021**, *11*, 6269.
43. Kang, H.; Zhou, H.; Wang, X.; Chen, C. Real-time fruit recognition and grasping estimation for robotic apple harvesting. *Sensors* **2021**, *20*, 5670. [CrossRef]
44. Huang, W.; Miao, Z.; Wu, T.; Guo, Z.; Han, W.; Li, T. Design of and Experiment with a Dual-Arm Apple Harvesting Robot System. *Horticulturae* **2024**, *10*, 1268. [CrossRef]
45. Xie, F.; Guo, Z.; Li, T.; Feng, Q.; Zhao, C. Dynamic Task Planning for Multi-Arm Harvesting Robots Under Multiple Constraints Using Deep Reinforcement Learning. *Horticulturae* **2025**, *11*, 88. [CrossRef]
46. Chen, W.; Xu, T.; Liu, J.; Wang, M.; Zhao, D. Picking robot visual servo control based on modified fuzzy neural network sliding mode algorithms. *Electronics* **2019**, *8*, 605. [CrossRef]
47. Zhang, K.; Lammers, K.; Chu, P.; Li, Z.; Lu, R. An automated apple harvesting robot—From system design to field evaluation. *J. Field Rob.* **2024**, *41*, 2384–2400. [CrossRef]
48. Zhang, Z. A flexible new technique for camera calibration. *IEEE Trans. Pattern Anal. Mach. Intell.* **2002**, *22*, 1330–1334. [CrossRef]
49. Lowe, D. G. Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vis.* **2004**, *60*, 91–110. [CrossRef]
50. Bay, H.; Ess, A.; Tuytelaars, T.; Van Gool, L. Speeded-up robust features (SURF). *Comput. Vis. Image Underst.* **2008**, *110*, 346–359. [CrossRef]
51. Li, T.; Fang, W.; Zhao, G.; Gao, F.; Wu, Z.; Li, R.; Dhupia, J. An improved binocular localization method for apple based on fruit detection using deep learning. *Inf. Process. Agric.* **2023**, *10*, 276–287. [CrossRef]
52. Zhang, G.; Tian, Y.; Yin, W.; Zheng, C. An apple detection and localization method for automated harvesting under adverse light conditions. *Agriculture* **2024**, *14*, 485. [CrossRef]
53. Hu, T.; Wang, W.; Gu, J.; Xia, Z.; Zhang, J.; Wang, B. Research on apple object detection and localization method based on improved YOLOX and RGB-D images. *Agronomy* **2023**, *13*, 1816. [CrossRef]
54. Zhang, H.; Tang, C.; Sun, X.; Fu, L. A refined apple binocular positioning method with segmentation-based deep learning for robotic picking. *Agronomy* **2023**, *13*, 1469. [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.