agriculture

MDPI

Article

Research on Three-Dimensional Positioning Method for Automatic Strawberry Fruit Picking Based on Vision–IMU Fusion

Bowen Liu [1], Chuhan Chen [1], Junqiu Li [1,*], Qinghui Zhang [1] and Yinghao Meng [2]

[1] College of Big Data and Intelligent Engineering, Southwest Forestry University, Kunming 650000, China; lbwen_swfu@163.com (B.L.); chenchuhan2005@163.com (C.C.); huizq@swfu.edu.cn (Q.Z.)
[2] Oak Ridge Innovation Institute, University of Tennessee, Knoxville, TN 37996, USA
* Correspondence: lijunqiu@swfu.edu.cn

Abstract

Accurate fruit localization and efficient harvesting are key challenges for agricultural robots, especially in dynamic orchard environments, where platform vibration, fruit occlusion, and computational resource limitations of embedded devices significantly impact system performance. To address these issues, this paper proposes a lightweight "fruit detection + harvesting" framework. First, by integrating MobileNetV4 and Triplet Attention mechanisms, an improved YOLOv8n network is designed, with the improved YOLOv8n Precision reaching 98.148% and FPS reaching 30 FPS on Jetson Nano, achieving a good balance between detection accuracy and computational efficiency suitable for edge deployment. Second, a strawberry three-dimensional coordinate reconstruction method based on weighted 3D centroid reconstruction is proposed, utilizing depth bias adjustment coefficients to improve spatial accuracy. Third, to address localization errors caused by vibration and platform motion, a dynamic compensation and temporal fusion strategy based on an Inertial Measurement Unit (IMU) is proposed. The rotation matrix estimated from IMU data is first used to correct camera pose variations. Then, an adaptive sliding window is employed to smooth the coordinate sequence. Finally, an Extended Kalman Filter (EKF) is applied to further refine the fused results by incorporating temporal dynamics, ensuring that the reconstructed three-dimensional coordinates in the robotic arm reference frame achieve higher stability and continuity. Experimental results in orchard scenarios show that compared with traditional methods, the system has higher localization accuracy, stronger robustness to dynamic disturbances, and higher harvesting efficiency. This work provides a practical and deployable solution for advancing intelligent fruit-harvesting robots.

Keywords: YOLOv8n; harvesting robot; Triplet Attention; MobileNetV4; weighted 3D centroid reconstruction; IMU assisted compensation

Check for updates

Received: 15 March 2026
Revised: 11 April 2026
Accepted: 14 April 2026
Published: 17 April 2026

Copyright: © 2026 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.

1. Introduction

China, as one of the world's largest fruit-producing countries [1–3], plays a key role in global agricultural production [4]. In recent years, rapid rural labor migration and rising labor costs have exposed the inefficiency and unsustainability of traditional manual harvesting systems [5–7]. This issue is particularly evident in high-value horticultural crops, such as greenhouse strawberries, tomatoes, and peppers, where dense planting and long production cycles further increase harvesting difficulty.

Although mechanical harvesting technologies have been explored, their application remains limited due to crop variability and complex canopy structures in real environments [8,9]. Therefore, developing reliable automated picking systems has become essential.

Achieving a complete pipeline from fruit detection to precise 3D localization and robotic picking still faces challenges in perception accuracy, localization stability, and real-time performance under complex orchard conditions [10,11]. Existing studies have widely adopted deep learning combined with optimized localization strategies.

However, during fruit detection, large model size and high computational demand hinder deployment on edge devices. To address this, various improvements have been proposed. For example, Zhu et al. (2024) [12] introduced YOLOv5s-CEDB by integrating CoordAtt, DConv, and EVC modules, improving detection performance in complex environments, but at the cost of increased model complexity. Li et al. [13] adopted ShuffleNetV2 with BiFPN for multi-scale fusion, improving inference speed but still failing to balance accuracy and compactness.

To further reduce computational cost, Teng et al. (2025) [14] proposed DS-YOLO, which simplifies network topology but weakens feature representation in complex scenes. Similarly, An et al. (2022) [15] designed SDNet based on YOLOX, improving robustness to occlusion, yet its real-time performance on resource-limited platforms remains insufficient. Overall, existing methods still struggle to balance lightweight design and detection performance.

In addition, detection is only the first step in the harvesting pipeline. Accurate 3D localization is critical for reliable robotic grasping. Most existing methods follow a pipeline of detection, depth acquisition, and coordinate transformation, typically relying on mapping the bounding box centroid to depth data. However, under occlusion, irregular fruit shapes, or illumination changes, this mapping becomes unreliable, leading to significant localization errors.

To address this issue, some studies attempt to directly predict optimal grasping points. For instance, Su et al. (2025) [16] proposed AP-UNet, incorporating ASPP and hierarchical feature enhancement for improved detection. Ma et al. (2025) [17] developed STRAW-YOLO based on YOLOv8-pose, using keypoint detection to reduce localization errors. Similarly, Li et al. [18] proposed StarBL-YOLO, combining RGB-D sensing for robust multi-target grasping point detection.

However, these approaches often increase computational cost due to more complex architectures. Zhao et al. [19] introduced YOLOv7-hv with a mask branch for 6D pose estimation, improving 3D perception but limiting deployment due to large model size. Moreover, most studies are conducted under static conditions. Although Mei et al. (2025) [20] explored 3D vision-based localization, dynamic disturbances such as vibration remain insufficiently addressed. Rana et al. (2026) [21] further emphasized the importance of environmental robustness for autonomous systems.

To find a balance between precise positioning and lightweight computation, some scholars have begun exploring alternative solutions. For example, Liu et al. employed HSV color thresholding to determine measurement points [22], then used a fixed-diameter model to back-calculate the fruit center. Although the recognition success rate could reach 95% in locally occluded scenarios, the thresholds required manual calibration for each variety, and errors accumulated rapidly under conditions of fruit overlap or long-distance observation, revealing inherent flaws such as "sensitivity to prior parameters and poor scene adaptability." Yang et al. [23] further constrained the YOLOv8s detection bounding box, using the mode depth within a 0.7-fold sub-rectangle as the center to mitigate interference from background noise outside the bounding box. However, this method still assumed uniform and reliable depth maps within the box; when dense vines caused depth loss or an increase in outliers, the centroid depth deviated systematically from the actual surface, leading to a decline in harvesting success. Xiao et al. [24] introduced Detect-COF-guided SGBM stereo matching, limiting the matching search to the bounding box and supplementing it with Sobel edge constraints, reducing the relative error to 2.6%

within a 20 cm working distance and significantly enhancing positioning robustness in occluded scenes. However, its performance heavily relied on a fixed baseline length, and the problem of parallax map degradation remained inevitable when fruit distance increased or surface texture was sparse.

In practical scenarios, vibrations from robotic arms and mobile platforms further degrade localization stability. On resource-constrained embedded devices, detection speed is limited, and motion-induced jitter leads to unstable coordinates, making it difficult to rely solely on visual and depth information.

To address these challenges, this study proposes a framework combining pixel-weighted centroid reconstruction with vision–IMU fusion. First, YOLOv8 is adopted as the baseline due to its mature architecture and stable deployment performance [25]. A lightweight detection model is developed by replacing the backbone with MobileNetV4 and introducing Triplet Attention, reducing model size while maintaining accuracy. Second, valid depth pixels within the bounding box are extracted, and a weighted fusion strategy based on spatial proximity and depth continuity is applied to obtain more reliable 3D coordinates. Finally, IMU measurements are used to estimate motion states and compensate for dynamic disturbances. An adaptive weighting strategy and EKF are further employed to improve temporal consistency and robustness.

To validate the proposed system, comprehensive experiments were conducted. The improved detection model was compared with mainstream methods, and the proposed 3D reconstruction method was evaluated through parameter optimization. The robustness of IMU-based compensation was tested under different vibration conditions. In addition, ablation studies were performed to analyze the contributions of individual components. Finally, experiments in both real orchard environments (Yunchuang Farm, Panlong District, Kunming, Yunnan Province) and laboratory settings verified the effectiveness and applicability of the system. The main contributions of this study are summarized as follows:

1. To improve the real-time performance of object detection running on embedded systems, this paper proposes a lightweight fruit detection model based on the YOLOv8n model, balancing accuracy and deployment performance. The paper introduces the MobileNetV4 backbone network and the Triplet Attention mechanism, achieving high-precision and high-efficiency fruit target detection on edge computing platforms. While maintaining a small model parameter count, both detection accuracy and inference speed outperform existing similar models, making it suitable for resource-constrained agricultural robot application scenarios.
2. To address the issue of significant errors in traditional bounding box center point localization under occlusion and complex background conditions, this paper designs a multi-factor weighted 3D centroid reconstruction method for fruits. The paper constructs a 3D coordinate-weighted reconstruction mechanism that integrates position weights and depth continuity weights. This method effectively enhances the accuracy and stability of fruit spatial localization, particularly demonstrating stronger robustness in occluded environments.
3. To tackle the issue of 3D localization instability caused by dynamic disturbances such as robotic arm vibrations in orchard environments, this paper introduces an IMU-based dynamic compensation and temporal fusion strategy. By integrating acceleration and angular velocity measurements, a motion state estimation mechanism is established to characterize platform dynamics in real time. Under vibration, motion-induced deviations are first compensated through IMU-based pose correction. Subsequently, an adaptive weighting strategy is employed within a sliding window to suppress unreliable observations, where frames affected by strong disturbances are assigned lower confidence. The fused results are further refined using an Extended

Kalman Filter (EKF), enabling temporal consistency through dynamic modeling. This cascaded framework significantly reduces spatial jitter in the output and enhances the continuity and reliability of 3D localization.

2. Materials and Methods

2.1. Overview of Fruit Precise Positioning Research

This paper proposes a method for precise and stable positioning of fruits in complex orchard environments and integrates it into an automatic harvesting robot. The overall process includes three stages—fruit detection, 3D coordinate reconstruction, and IMU-assisted dynamic compensation—forming a complete closed-loop framework from perception to decision-making to execution, as shown in Figure 1. First, in the fruit detection stage, this paper constructs a lightweight detection network based on YOLOv8n and introduces the MobileNetV4 backbone structure and Triplet Attention mechanism, effectively improving feature extraction capabilities and detection accuracy while significantly reducing parameter size and computational load, enabling the model to be efficiently deployed on resource-constrained agricultural robot platforms. Second, in the 3D coordinate reconstruction stage, addressing the issue of traditional methods relying on bounding box centers and depth maps for direct mapping, which are easily affected by noise and occlusion, this paper proposes a weighted centroid-based reconstruction method. Within the detection box, valid depth pixels are selected, and a weighted averaging method based on spatial proximity and depth continuity is employed to reduce the influence of occlusion and abnormal depth values, resulting in a more accurate estimation of the fruit's 3D geometric center. Subsequently, under dynamic interference conditions, platform vibration and robotic arm movement cause significant jitter in the reconstructed coordinates.

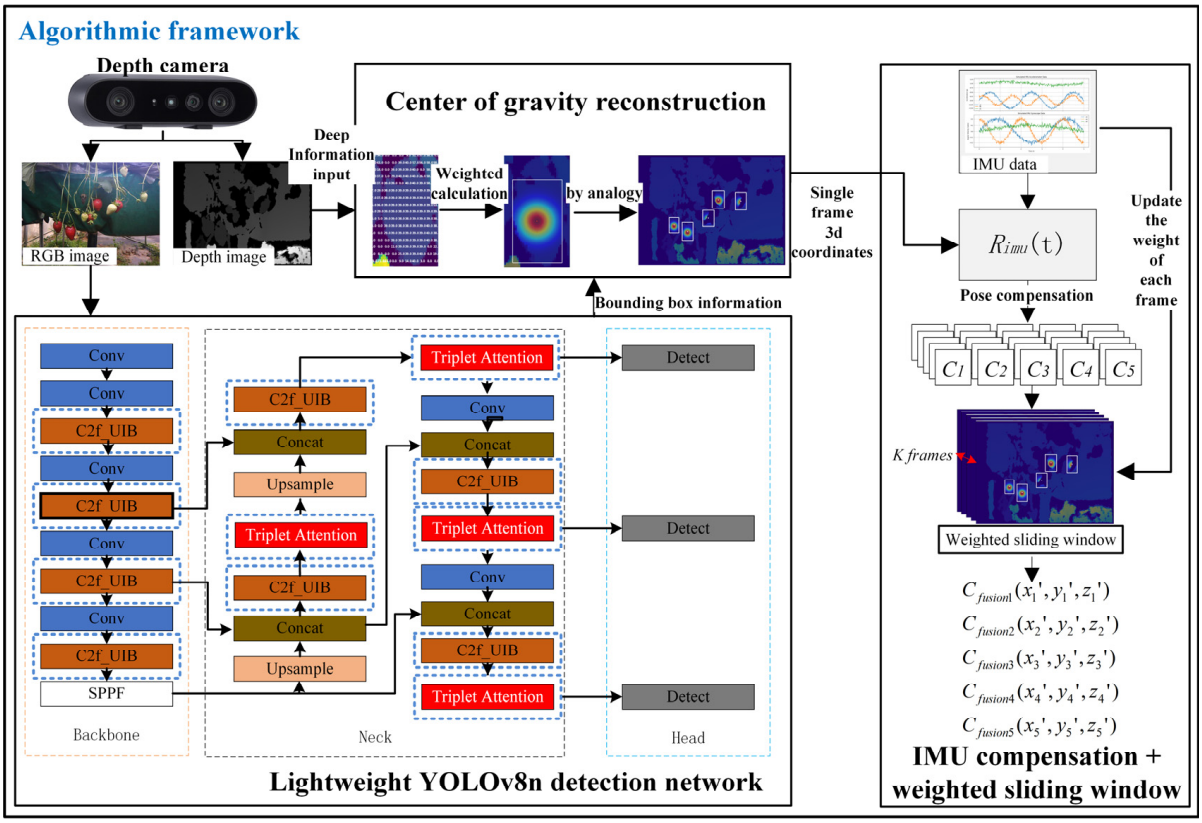


Figure 1. Algorithm framework.

To this end, an IMU-based dynamic compensation mechanism is designed. The rotation matrix estimated in real time from IMU data is used to correct coordinate offsets caused by platform motion. An adaptive sliding window fusion strategy is then applied to suppress transient disturbances by dynamically adjusting observation weights. Furthermore, an EKF is introduced to refine the fused results by incorporating temporal dynamics, thereby effectively reducing spatial jitter and improving the continuity and robustness of 3D localization.

The fruit harvesting robot system designed in this paper adopts a modular design, enhancing the maintainability of the robot system and leaving development interfaces for modules yet to be addressed to facilitate future functional expansion. As shown in Figure 2, the overall harvesting robot system is divided into four modules: the algorithm main control Jetson nano module, the motion control main control STM32 control module, the sensor module, and the robotic arm. Among them, the camera adopts a Gemini binocular structured-light design, which supports depth detection up to 2.5 m at 0.25~ and has a resolution of 640 × 480 pixels to ensure clear images for easy transmission and processing. An H30 inertial measurement unit (IMU) developed by Wheeltec is employed to provide real-time motion and orientation information. The sensor integrates a 9-axis MEMS system, including a three-axis gyroscope, accelerometer, and magnetometer, and supports a high output frequency of up to 400 Hz. According to the manufacturer's specifications, the IMU achieves attitude measurement accuracy of approximately 0.1° for roll and pitch, and within 1° for heading under magnetic assistance, with low bias drift and noise characteristics. The robotic arm uses a five-degree-of-freedom SCARA robotic arm, leveraging its multi-degree-of-freedom advantages to excellently complete the harvesting task of target fruits.

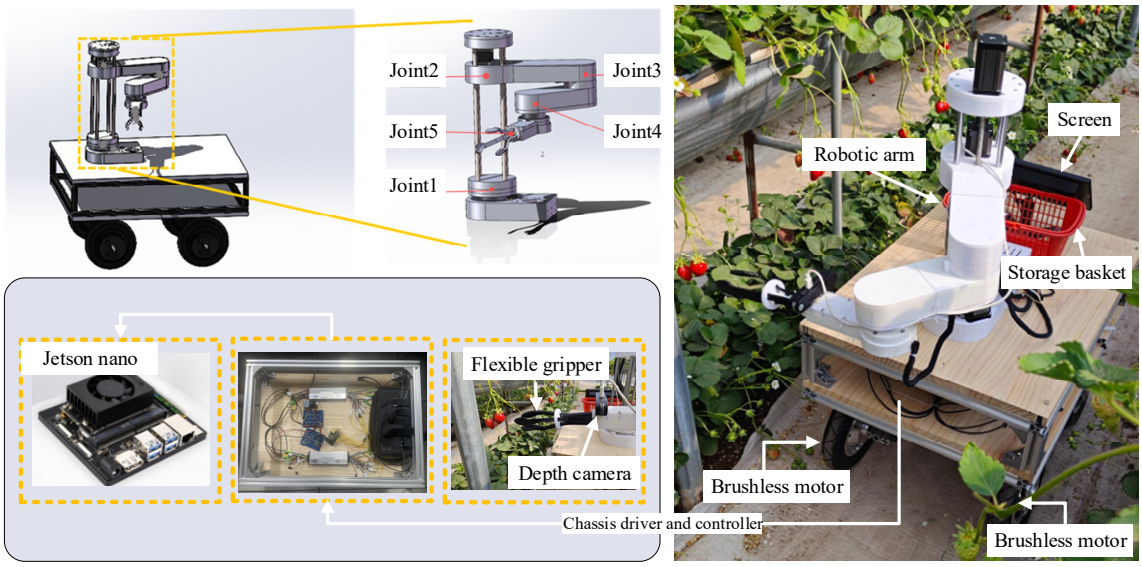


Figure 2. Photo of a strawberry-picking vehicle.

2.2. The Establishment of the Dataset

The dataset was collected at Yunchuang Farm in Kunming, Yunnan Province, China, during the 2024 growing season. In this study, strawberry fruits were selected as the experimental objects. To ensure consistency between the visual perception of the robot during actual harvesting and the experimental data, all images were collected using the Gemini Pro camera mounted on the harvesting robot. The images were stored in JPEG format with a resolution of 640 × 480 pixels. Data collection was conducted at Yunchuang Farm in Kunming, Yunnan Province, China, where a total of 800 images of mature strawberries were acquired. The dataset covers a wide range of realistic orchard conditions,

including backlighting and front lighting, as well as occlusion caused by branches, leaves, and immature fruits.

Existing studies on strawberry detection have constructed datasets with different focuses, such as occlusion-level annotation [26], multi-stage growth classification [15], or lightweight deployment scenarios [14]. In a similar manner, the dataset in this study is designed to reflect practical harvesting conditions, emphasizing the coexistence of illumination variation and complex occlusion in real environments.

After data acquisition, all original images were manually annotated using LabelImg to generate bounding-box labels for fruit detection. In the data preprocessing stage, the original dataset was first randomly split into training, validation, and test sets to ensure fair performance evaluation under real-world conditions. Subsequently, data augmentation was applied only to the training set to increase sample diversity and improve model generalization. The augmentation operations included affine transformations, elastic deformations, perspective transformations, Gaussian noise injection, and the simulation of environmental effects such as rain, snow, and sun flare.

After augmentation, the training set was expanded to 5760 images, while the validation and test sets contained 1920 images each, resulting in a final dataset of 9600 images for model training and evaluation.

2.3. YOLOv8 Network Improvement Strategy

YOLOv8n is the lightest and fastest running model in the YOLOv8 series, specifically designed for fast inference tasks on resource-constrained devices. Unlike previous YOLO versions, YOLOv8 adopts a completely new network architecture, removing traditional anchor boxes in favor of an anchor-free design, which simplifies the training process and enhances the model's generalization ability. YOLOv8n retains core object detection capabilities while significantly compressing model size and computational complexity, making it highly suitable for deployment on edge computing devices [27]. YOLOv8n typically contains fewer than 3 million parameters, exhibits extremely low latency, and can achieve real-time or near-real-time object detection and localization tasks while maintaining reasonable detection accuracy. The model has been validated on standard datasets like COCO, demonstrating a good balance between speed and accuracy, and is widely used in performance-sensitive visual scenarios such as drone vision, mobile robotics, and automated harvesting [28].

To reduce the parameter count and weight file size of YOLOv8 for easier deployment in embedded systems, this study replaces the backbone network of YOLOv8n with the lightweight MobileNetV4 network structure and adds a Triplet Attention module to improve the model's detection accuracy when handling densely arranged fruits and partially occluded objects.

MobileNetV4 [29–31] effectively reduces the model's parameter count and weight file size by introducing the universal inverted bottleneck (UIB) architecture component. The UIB structure adds two optional depthwise convolutions (Optional DWConv) to the traditional inverted bottleneck (IB) structure—one before the expansion layer and one between the expansion layer and the projection layer. As shown in the UIB structure, placing depthwise convolutions (DepthWise) at different positions between the expansion layer and the projection layer yields four configurations, as illustrated in Figure 3. The feedforward network (FFN) places no DepthWise convolutions, representing the most basic accelerator; the Conv Next places a DepthWise convolution between the expansion layer and the projection layer, achieving larger kernel sizes; IB places a DepthWise convolution before the expansion layer, enabling greater model capacity; and the Extra DW structure places DepthWise convolutions both before the expansion layer and between the expansion

layer and the projection layer, combining the advantages of Conv Next and IB. At each network stage, the UIB structure provides sufficient flexibility to achieve temporary spatial and channel mixing trade-offs, expanding receptive fields, and maximizing computational efficiency as needed.

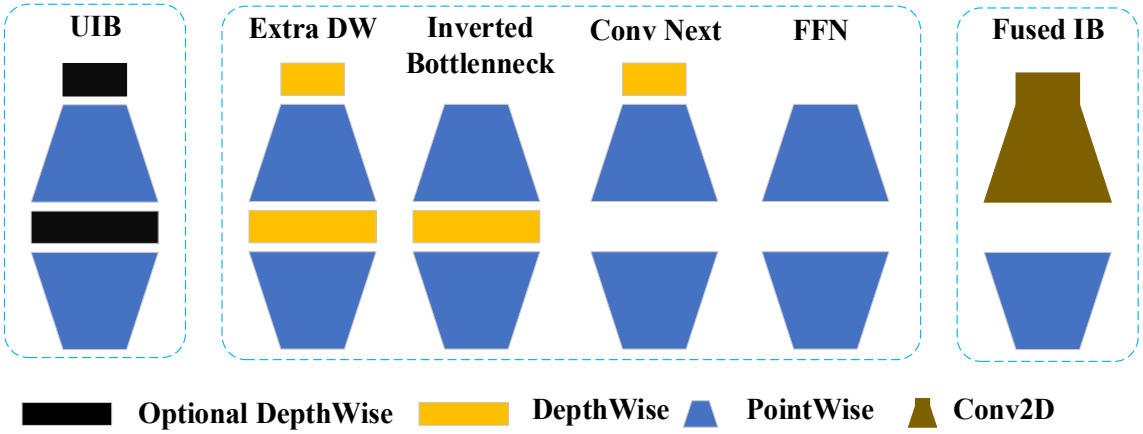


Figure 3. Diagram of UIB and other modules.

2.3.1. Triplet Attention Module

To enable the model to find key information segments in vast information streams, adding an attention mechanism can effectively improve the model's processing efficiency and effectiveness. The attention mechanism dynamically adjusts parameters, allowing the model to perceive information more finely while helping the model better understand the relationships between the upper and lower ends of the information stream.

Common attention mechanisms increase the network's computational complexity and memory consumption when processing large-scale image data, and traditional channel attention mechanisms lack information interaction. The Triplet Attention mechanism proposed by Misra D et al. can effectively solve this problem. Triplet Attention achieves cross-dimensional interaction between spatial dimensions and image channel dimensions through three branches, featuring parameter-free lightweight characteristics that reduce computational complexity and memory consumption while preserving more spatial information. The three branches of Triplet Attention are shown in Figure 4.

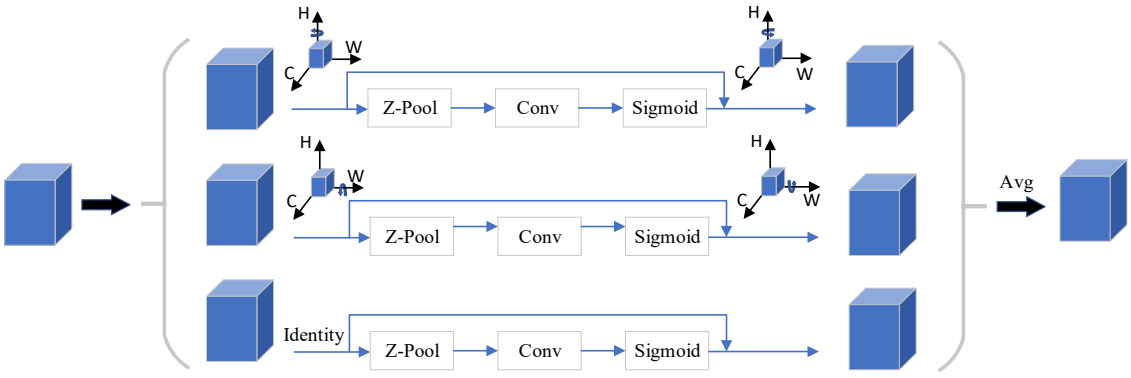


Figure 4. Triplet Attention schematic diagram.

During the transformation process, the Z-pool layer reduces the tensor in dimension C to 2D and concatenates the average pooling features and maximum pooling features along this dimension. After calculating the attention weights for the three branches separately, the tensors generated by these three branches are merged together by taking the average.

The specific calculation method is shown in Figure 5, resulting in the final tensor being Formula (1), where σ represents the activation function Sigmoid, and ψ_1, ψ_2, ψ_3 represents the standard convolutional layer of 7×7.

$$y = \frac{1}{3}\left\{\overline{M_1\sigma\left[\psi_1\left(M_1'\right)\right]} + \overline{M_2\sigma\left[\psi_2\left(M_2'\right)\right]} + M_3\sigma\left[\psi_3\left(M_2'\right)\right]\right\} \quad (1)$$

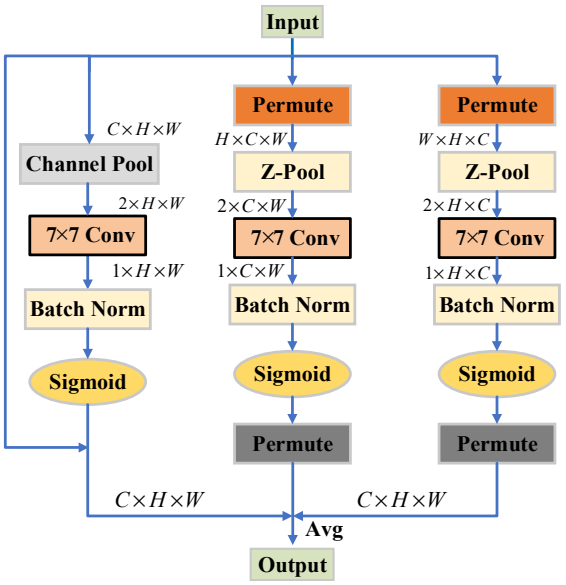


Figure 5. Triplet Attention computation flowchart.

2.3.2. Improved YOLOv8 Model Structure

YOLOv8n is the model with the fewest parameters in the YOLOv8 series. To meet the high real-time performance requirements of the YOLOv8 model in embedded systems, this study adopts the YOLOv8n object detection model to establish the detection system. At the same time, the original YOLOv8n is improved to better align with the high precision and high-speed requirements of the proposed harvesting robot. The backbone network is replaced with MobileNetV4, and the C2f module is replaced with C2f_UIB. A Triplet Attention module is added after the C2f_UIB module in the Neck structure to improve detection accuracy. The improvements are shown in the blue box of Figure 6.

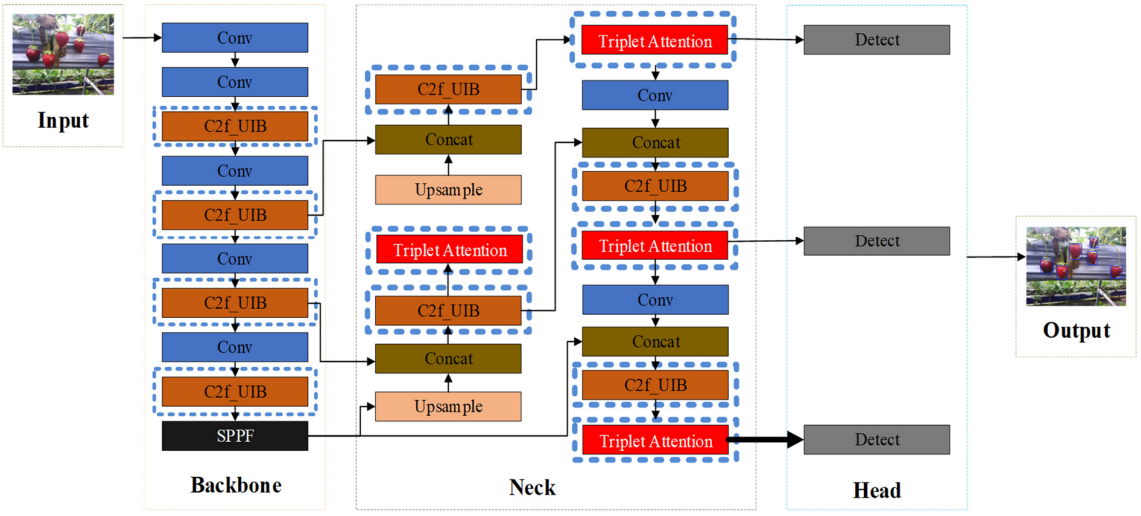


Figure 6. Schematic diagram of the improved YOLOv8n model structure.

2.4. Coordinate System Transformation

To achieve precise fruit picking by the robotic arm, it is necessary to convert the three-dimensional coordinates of the fruit in the camera coordinate system to the robotic arm coordinate system. This paper first uses the improved YOLOv8n model to identify fruits in the image and calculates the weighted centroid coordinates (u, v) of the fruit in the camera coordinate system based on the pixel-weighted bounding box method. Subsequently, by extracting the depth information Z_C of the corresponding pixels from the depth map, the three-dimensional position of the fruit in the camera coordinate system is obtained.

As shown in Figure 7c, the transformation between the world coordinate system (X_W, Y_W, Z_W) and the camera coordinate system (X_C, Y_C, Z_C) can be achieved through rotation matrices R and translation vectors T. Assuming the rotation matrix is R and the translation vector is T, the coordinates of point P in the camera coordinate system can be obtained using Equation (2):

$$\begin{bmatrix} X_C \\ Y_C \\ Z_C \end{bmatrix} = R \begin{bmatrix} X_W \\ Y_W \\ Z_W \end{bmatrix} + T \tag{2}$$

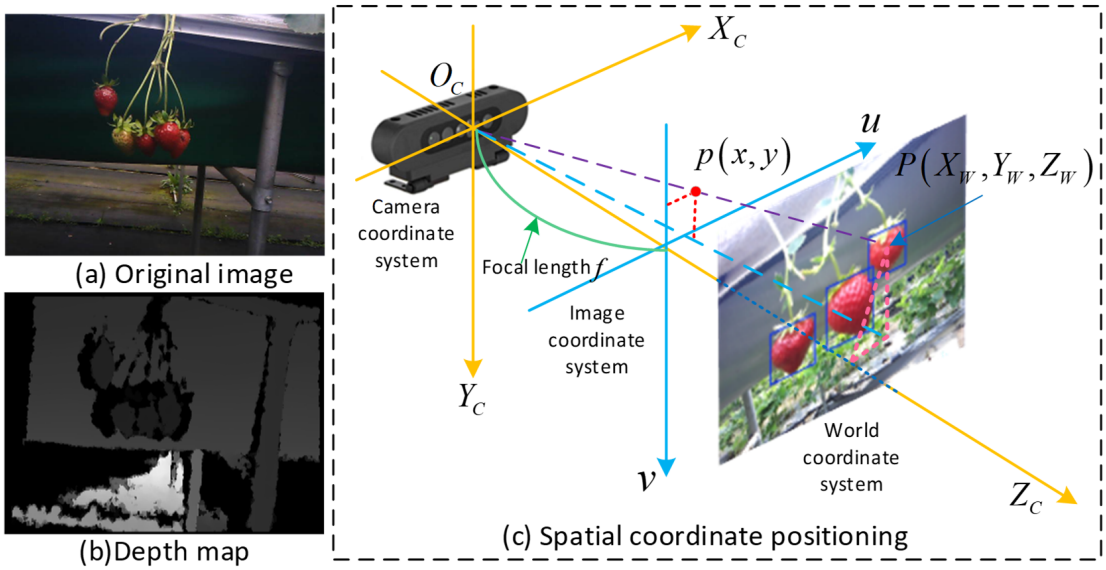


Figure 7. Schematic diagram of coordinate system relationships.

Substituting the relevant parameters into Equation (2) yields the complete projection model. By implementing Zhang's calibration method, the specific values of the camera's internal parameters were determined. As shown in Equation (3), the focal lengths f_x and f_y were calculated as 452.54 and 451.93, respectively, while the principal point coordinates (u_0, v_0) were determined to be (326.37, 239.60):

$$Z_C \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} 452.54 & 0 & 326.37 \\ 0 & 451.93 & 239.60 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} R_{3\times3} \\ T_{3\times1} \end{bmatrix} \begin{bmatrix} X_W \\ Y_W \\ Z_W \\ 1 \end{bmatrix} \tag{3}$$

To ensure geometric accuracy, the distortion coefficients D were also calibrated as [0.06187, −0.05180, −0.00433, 0.00034, −0.10323] to compensate for lens-induced distortions. The resulting average reprojection error was 0.03705 pixels, demonstrating the high precision of the calibration process.

To achieve spatial alignment of the camera coordinate system and the robotic arm coordinate system, a hand-eye calibration method based on a calibration plate was adopted, as shown in Figure 8b. To further enhance the calibration accuracy, a probe (Figure 8a) was installed on the end effector of the robotic arm to precisely determine the position of the end reference point and obtain more accurate calibration data. By calibrating the internal and external parameters of the camera, the spatial coordinates of strawberries in the camera coordinate system can be converted into the robot coordinate system, thus ensuring that the robot can accurately pick strawberries [32].

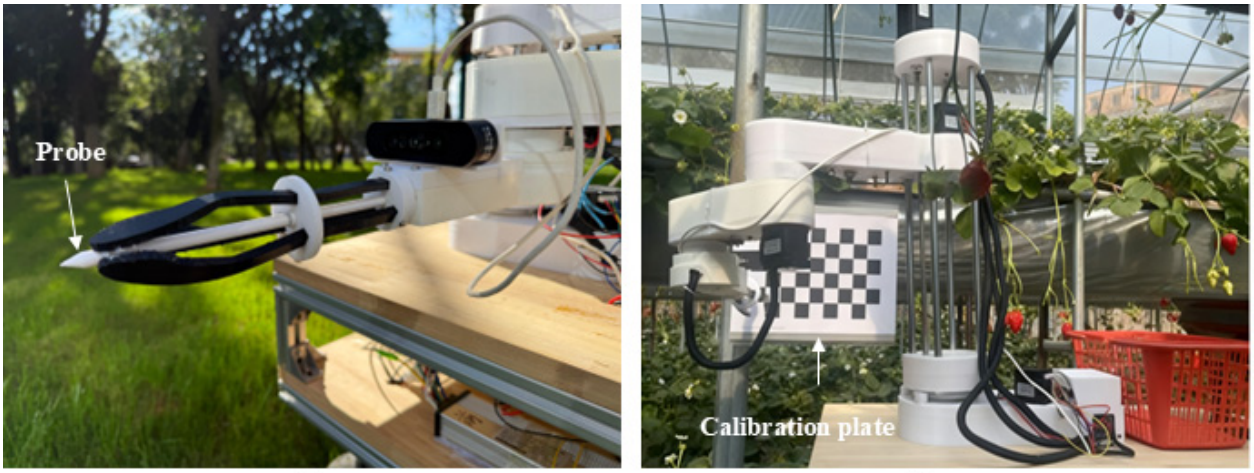


Figure 8. Hand-eye calibration diagram. (**a**) The auxiliary probe installed on the end effector; (**b**) Hand-eye calibration schematic diagram.

2.5. Weighted Geometric Centroid Reconstruction of Fruit

To achieve high-precision grasping of fruits by a robotic arm, accurately obtaining the position coordinates of fruits in three-dimensional space is crucial. Traditional methods typically rely on the center point of two-dimensional bounding boxes output by target detection models, combined with the depth value of corresponding pixels in the depth map, to calculate the three-dimensional coordinates of fruits in the camera coordinate system. However, the center point of bounding boxes often only reflects the two-dimensional projection center and is affected by factors such as occlusion and irregular fruit shapes, leading to a deviation between the center point and the actual spatial centroid.

For this purpose, this paper proposes a three-dimensional coordinate reconstruction method based on weighted centroid of bounding box pixels. First, the system uses an improved YOLOv8n model to perform object detection on the input image, obtaining the two-dimensional bounding box position of fruits in the image coordinate system. Simultaneously, a depth map is acquired through a binocular structured-light camera, and a subset of depth information corresponding to the bounding box area is extracted. Assuming the detection bounding box of the target fruit in the image coordinate system is

$$B = \{(u_{min}, v_{min}), (u_{max}, v_{max})\} \quad (4)$$

Among them, (u_{min}, v_{min}) and (u_{max}, v_{max}) are the pixel coordinates of the upper-left and lower-right corners of the bounding box, respectively.

The center point of the bounding box is defined as

$$(u_c, v_c) = \left(\frac{u_{min} + u_{max}}{2}, \frac{v_{min} + v_{max}}{2} \right) \tag{5}$$

Project the pixel coordinate (u_i, v_i, d_i) back to the 3D position in the camera coordinate system:

$$X_i = \frac{(u_i - c_x) \cdot d_i}{f_x}, Y_i = \frac{(v_i - c_y) \cdot d_i}{f_y}, Z_i = d_i \tag{6}$$

where (c_x, c_y) is the principal point coordinate of the camera, and (f_x, f_y) is the camera focal length (in pixels).

Secondly, for each pixel, a weight coefficient w_i is assigned based on its spatial position and depth continuity. The weight design in this paper comprehensively considers the spatial position of the centroid and the continuity of depth values, meaning pixels closer to the center of the bounding box have higher weights, and pixels with smoother and more reliable depth values also have higher weights.

$$w_i^{pos} = \exp\left(-\frac{(u_i - u_c)^2 + (v_i - v_c)^2}{2\sigma^2} \right) \tag{7}$$

$$w_i^{depth} = \exp\left(-\frac{\left| d_i - \bar{d} \right|}{\kappa} \right) \tag{8}$$

$$w_i = w_i^{pos} \cdot w_i^{depth} \tag{9}$$

Among them, σ is a hyperparameter that controls the rate of weight decay, typically set adaptively based on the size of the bounding box. $\bar{d}$ represents the mean depth of all valid pixels within the bounding box area, and κ is the depth deviation adjustment coefficient. The decay coefficient σ for spatial position weights is set based on the width $W = u_{max} - u_{min}$ of the bounding box:

$$\sigma = \frac{W}{4} \tag{10}$$

The depth deviation adjustment coefficient κ is set based on the empirical accuracy of the camera's depth measurement, with a value of 25 mm.

Based on the above weights, the weighted spatial centroid coordinates of the fruit are calculated:

$$\left(x_g, y_g, z_g \right) = \left(\frac{\sum_i w_i u_i}{\sum_i w_i}, \frac{\sum_i w_i v_i}{\sum_i w_i}, \frac{\sum_i w_i d_i}{\sum_i w_i} \right) \tag{11}$$

$\left(x_g, y_g, z_g \right)$ is the spatial coordinate of the final weighted centroid reconstruction of the fruit. As shown in Figure 9, the red points are the centroid coordinates after reconstruction, and the white points are the center points of the bounding box. It can be seen that when the center point of the bounding box falls on an occluder, the corresponding depth value will take the depth of the occluder, resulting in errors, while the reconstructed coordinates fall on the strawberry fruit. This method can still obtain relatively accurate positioning points on the fruit even in the presence of occlusions.

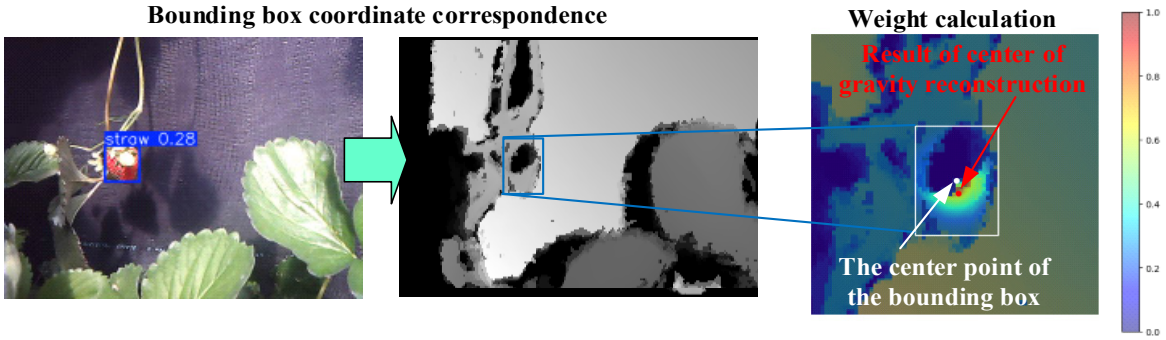


Figure 9. Centroid reconstruction result.

2.6. IMU Dynamic Compensation and Data Fusion

In fruit 3D localization, visual-depth spatial reconstruction often assumes the camera platform stability. However, in real orchards, the vibration of the robotic arm base, slight displacement at the end, and abrupt stop offset during movement introduce two types of errors: one is the geometric pose deviation caused by camera attitude jitter, and the other is the depth measurement noise caused by instantaneous mismatch of the depth sensor or light interference.

As illustrated in Figure 10, platform motion mainly manifests as rigid-body rotation and translation of the fruit point cloud in the camera coordinate system. This effect can be corrected using real-time attitude information obtained from the IMU, where the rotation matrix is applied to compensate for instantaneous camera pose variations, allowing the observed points to be transformed into a stable reference frame.

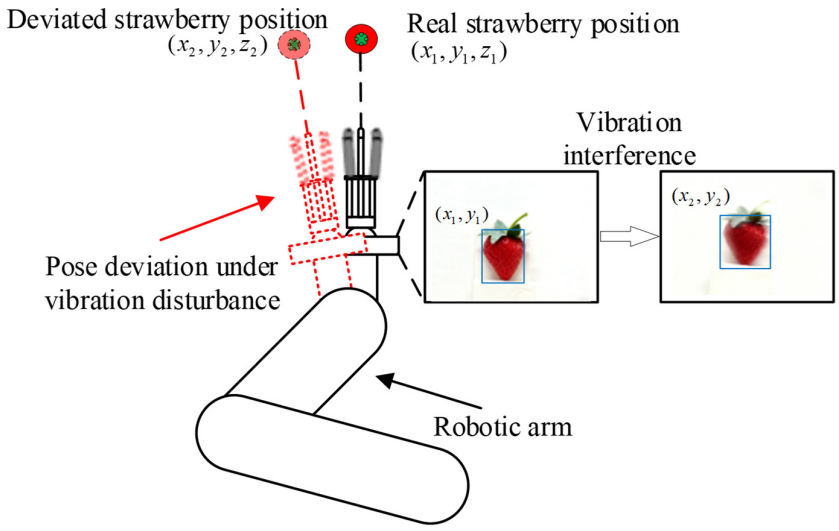


Figure 10. The geometric pose deviation caused by the overall posture shake of the camera.

However, IMU-based compensation alone cannot address depth measurement noise. To overcome this limitation, an adaptive sliding window mechanism is introduced to perform multi-frame fusion. Specifically, observations from frames affected by severe vibration are assigned lower weights, while stable measurements are emphasized, thereby reducing the influence of outliers and improving spatial consistency. To further enhance temporal stability, an EKF is employed to refine the fused coordinates by modeling the temporal evolution of fruit positions. This additional step effectively suppresses residual fluctuations and ensures smoother coordinate transitions.

By combining IMU-based pose compensation, sliding window fusion, and EKF-based refinement, the proposed method simultaneously mitigates attitude drift and depth-related

instability. It is worth noting that in cases of severe interference leading to target loss (e.g., failure to detect strawberries), the system triggers the robotic arm to return to its initial position for reinitialization, enabling recovery of the field of view. With the improved YOLOv8n ensuring robust target detection, the overall method can be summarized into four main steps.

First, correct the pose based on IMU information when ensuring the robotic arm is not performing planned motion, addressing the overall coordinate offset issue. The six-axis IMU on the platform can output three-axis angular velocity and three-axis acceleration in real time:

$$\omega_t = [\omega_x, \omega_y, \omega_z], a_t = [a_x, a_y, a_z] \tag{12}$$

Based on the angular velocity and acceleration output by the IMU, Euler angles (yaw ψ, pitch θ, roll ϕ) are obtained through attitude calculation. The rotation matrix is defined using the Z-Y-X sequence:

$$R_{imu}(t) = R_z(\psi) \cdot R_y(\theta) \cdot R_x(\phi) \tag{13}$$

where

$$R_z(\psi) = \begin{bmatrix} \cos\psi & -\sin\psi & 0 \\ \sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{bmatrix}, R_y(\theta) = \begin{bmatrix} \cos\theta & 0 & \sin\theta \\ 0 & 1 & 0 \\ -\sin\theta & 0 & \cos\theta \end{bmatrix}, R_x(\phi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\phi & -\sin\phi \\ 0 & \sin\phi & \cos\phi \end{bmatrix} \tag{14}$$

The composite rotation matrix is

$$R_{imu}(t) = \begin{bmatrix} \cos\psi\cos\theta & \cos\psi\sin\theta\sin\phi - \sin\psi\cos\phi & \cos\psi\sin\theta\cos\phi + \sin\psi\sin\phi \\ \sin\psi\cos\theta & \sin\psi\sin\theta\sin\phi + \cos\psi\cos\phi & \sin\psi\sin\theta\cos\phi - \cos\psi\sin\phi \\ -\sin\theta & \cos\theta\sin\phi & \cos\theta\cos\phi \end{bmatrix} \tag{15}$$

This matrix is used to correct the three-dimensional coordinates of the fruit in the camera coordinate system, compensating for attitude changes caused by platform vibration or tilt.

Using this matrix to compensate for the observed point P_C in the camera coordinate system:

$$P'_{CAM}(t) = R_{imu}(t) \cdot P_{CAM}(t) \tag{16}$$

This step ensures that even if the platform experiences slight tilting or rotation, the fruit remains approximately aligned with its actual physical position in the corrected coordinate system, avoiding systematic drift.

Step two, to apply the compensated observation results to the robotic arm grasping, the $P'_{CAM}(t)$ needs to be converted to the robotic arm base coordinate system $\{M\}$. The 3D position of the fruit in the base coordinate system is obtained through the following coordinate chain: first, transform the point $P'_{CAM}(t)$ in the camera coordinate system to the IMU coordinate system, then use the external parameter matrix $T_{IMU \leftarrow CAM}$ between the camera and IMU and the external parameter matrix $T_{EE \leftarrow IMU}$ between the IMU and end effector to map it to the end effector coordinate system; finally, through the forward kinematic solution obtained via the transformation matrix $T_{BASE \leftarrow EE}(t)$ from the end effector to the base, unify the point to the base coordinate system, thus obtaining

$$P_{BASE} = T_{BASE \leftarrow EE}(t) \cdot T_{EE \leftarrow IMU} \cdot T_{IMU \leftarrow CAM} \cdot P'_{CAM}(t) \tag{17}$$

This step achieves cross-modal coordinate unification, allowing the 3D position of the fruit to be directly used for robotic arm path planning and control.

Step three, although attitude correction and external parameter transformation can eliminate most of the systematic errors caused by dynamic interference, due to sensor measurement noise and environmental light changes, random fluctuations may still occur in single-frame results. On this basis, this paper introduces a sliding window of length K to store the corrected coordinates $\{P_1, P_2, \ldots, P_K\}$ of the recent K frames. To suppress the impact of dynamic interference, this paper designs a dynamic adaptive weight mechanism based on IMU status.

The weight of each frame is determined by the normalized intensity of acceleration and angular velocity:

$$\delta_i = \max\left(\frac{\|a_i\|_2}{\theta_a}, \frac{\|\omega_i\|_2}{\theta_\omega}\right) \tag{18}$$

$$\alpha_i = \exp(-\lambda \cdot \delta_i) \tag{19}$$

where $\|a_i\|_2 = \sqrt{a_{ix}^2 + a_{iy}^2 + a_{iz}^2}$ (L2 norm of the acceleration vector of the i frame, reflecting acceleration intensity); $\|\omega_i\|_2 = \sqrt{\omega_{ix}^2 + \omega_{iy}^2 + \omega_{iz}^2}$ (L2 norm of the angular velocity vector of the i frame, reflecting angular velocity intensity). a_i and ω_i are the acceleration and angular velocity of the i frame, respectively; θ_a, θ_ω is the threshold used to judge the dynamic level of the system, among which, $\theta_a = 1\ m/s^2$ (acceleration dynamic threshold, set based on actual vibration measurements of orchard robotic arms), $\theta_\omega = 5\ rad/s$ (angular velocity dynamic threshold, to avoid misjudgment of sudden stops and other violent movements); λ is the adjustment parameter, controlling the weight decay rate.

The weights of each frame are obtained through normalization:

$$w_i = \frac{\alpha_i}{\sum_{j=1}^{K} \alpha_j} \tag{20}$$

The final fused coordinates are

$$\hat{P}_{BASE} = \sum_{i=1}^{K} w_i \cdot P_i \tag{21}$$

Step four, although the sliding window fusion improves robustness, it does not explicitly model the temporal dynamics of fruit motion. To further enhance temporal consistency and reduce residual jitter, an Extended Kalman Filter (EKF) is introduced to refine the fused coordinates.

The state vector is defined as

$$x_k = \left[x, y, z, v_x, v_y, v_z\right]^T \tag{22}$$

where (x, y, z) represents the 3D position of the fruit, and (v_x, v_y, v_z) denotes the velocity components.

A constant-velocity motion model is adopted:

$$x_k = F_{x_k - 1} + w_k \tag{23}$$

where F is the state transition matrix and w_k is the process noise.

The observation model is defined as

$$z_k = H_{x_k} + v_k \tag{24}$$

where $\mathbf{z}_k = [x, y, z]^T$ is the fused coordinate obtained from the sliding window, and $\mathbf{v}k$ represents measurement noise.

The EKF recursively performs prediction and update steps to produce the final optimized estimate P_{ekf}. By incorporating temporal dynamics, this step effectively smooths residual fluctuations and enhances the continuity of fruit localization results.

In summary, the proposed method forms a four-stage processing pipeline in dynamic environments: IMU-based pose compensation, coordinate transformation, IMU-guided sliding window fusion, and EKF-based temporal refinement.

The sliding window reduces the impact of abnormal observations, while the EKF further enforces temporal consistency through dynamic modeling. The combination of these modules significantly improves the accuracy, robustness, and stability of fruit localization in unstructured orchard environments.

3. Experiments and Discussion

3.1. Performance Testing of Fruit Detection and Recognition

3.1.1. Experimental Platform and Test Environment

Image data were collected at Yunchuang Farm in Kunming, Yunnan Province, China, between May and July 2024. The robotic harvesting experiments were conducted during the same period under greenhouse conditions. The experimental environment for training the model in this study is as follows: GPU RTX 3090, 24 GB of memory, Ubuntu 20.04 operating system, PyTorch 1.11.0 framework, and Python version 3.8. The training parameters are set as follows: the number of training iteration rounds is 300, the batch image size is 32, and the input image size is 640 × 480 pixels.

3.1.2. Ablation Experiments

To systematically evaluate the contribution of each proposed module, ablation experiments were conducted based on the YOLOv8n baseline model. Two individual modules and their combinations were progressively integrated into the baseline model and compared with the final Ours-YOLO model, with the results summarized in Table 1. GFLOPs (Giga Floating Point Operations) is adopted as a metric to measure the computational complexity of the model, representing the number of floating-point operations required for a single forward pass. To ensure a fair comparison, all experiments were conducted under identical hardware configurations and training settings.

Table 1. Comparison of ablation experiments using the improved YOLOv8n model.

Network	Size (MB)	mAP50 (%)	mAP50–95 (%)	Precision (%)	Recall (%)	GFLOPs	Params
YOLOv8n	5.98	98.37	68.75	97.64	96.26	8.1	3,005,843
YOLOv8n-C2f	5.99	98.02	68.34	97.56	95.76	8.2	3,006,443
YOLOv8n-C2fneck	6.00	98.44	68.91	97.56	95.65	8.3	3,006,643
YOLOv8n-SPPF	6.04	98.86	69.29	97.61	95.84	8.2	3,038,811
YOLOv8n-MobileNetV4	4.44	98.12	67.56	97.50	95.29	6.1	2,183,251
MobileNetV4-YOLOv8n-C2f	4.47	98.05	68.28	96.84	95.97	6.3	2,183,851
MobileNetV4-YOLOv8n-C2fneck	4.47	97.85	68.49	97.35	95.37	6.3	2,184,051
MobileNetV4-YOLOv8n-SPPF	4.45	98.56	69.43	97.66	95.04	6.2	2,183,451

The ablation results show that replacing the YOLOv8n backbone with MobileNetV4 significantly reduces model complexity, with the model size decreased by 1.54 MB, the number of parameters reduced to 2,183,251, and the computational cost lowered to 6.1 GFLOPs. Model lightweighting improves detection speed on embedded robotic platforms, thereby enhancing overall picking efficiency. However, this modification leads to a 1.19% decrease in mAP50–95 and results in missed detections of some strawberry targets. This phenomenon is attributed to the reduced feature representation capability of the lightweight backbone, especially in scenarios with dense object distribution and severe occlusion.

To address this issue, the Triplet Attention mechanism is introduced to enhance feature representation. Specifically, the attention module is embedded at three different positions: (1) after the SPPF module in the backbone, (2) after the C2f module in the neck, and (3) between the neck and the detection head, denoted as SPPF, C2fNeck, and C2f, respectively. The corresponding changes in mAP50–95 are +0.54%, +0.16%, and −0.41%, while the increases in parameters and GFLOPs remain marginal. The results demonstrate that integrating Triplet Attention after the SPPF module achieves the most effective performance improvement.

It is worth noting that, although the mAP metric improves after introducing the attention mechanism, both Precision and Recall decrease. This phenomenon results from the feature redistribution effect of the attention mechanism; it enhances responses in target regions and suppresses background noise, thereby improving localization quality, while simultaneously weakening responses to partially occluded or boundary targets, which leads to missed detections.

Therefore, the decrease in Precision and Recall reflects a trade-off between feature representation and information filtering rather than a degradation of overall model performance. Meanwhile, the improved model achieves an effective balance between lightweight design and detection performance, making it suitable for fruit grasping tasks in strawberry field environments.

3.1.3. Comparative Experimental Analysis

To systematically evaluate the performance of lightweight detection models, 11 representative network architectures were selected for comparison, including YOLOv5n, YOLOv6n, YOLOv8n, YOLOv10n, FasterNet-YOLOv8n, MobileNetV1-YOLOv8n, MobileNetV2-YOLOv8n, MobileNetV3-YOLOv8n, ShuffleNetV1-YOLOv8n [33], ShuffleNetV2-YOLOv8n [34], and YOLOv8n-MobileNetV4-SPP. The quantitative results are shown in Table 2. To further evaluate the detection performance under different confidence thresholds and analyze the trade-off between precision and recall, precision–recall (PR) curves were used. Meanwhile, the variation in mAP50–95 over training epochs was plotted, as shown in Figure 11. The results show that MobileNetV4-YOLOv8n-SPPF achieves a higher final mAP50–95 value during training. Meanwhile, its PR curve is closer to the upper-right corner, indicating more stable performance and a better balance between precision and recall.

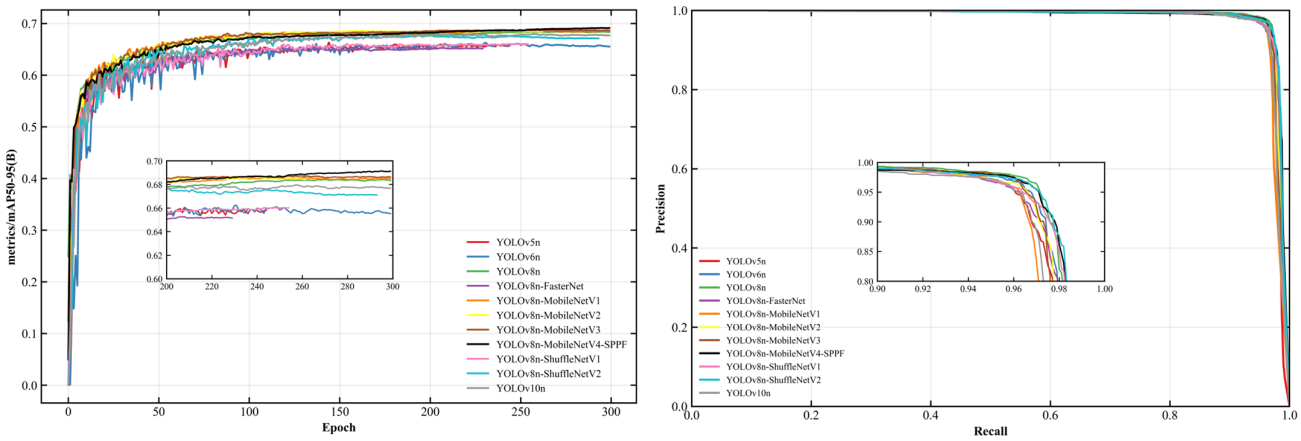


Figure 11. mAP50–95 curves and PR curves.

Table 2. Comparison Results of Various Improved Algorithms.

Network	Size (MB)	mAP50 (%)	mAP50–95 (%)	Precision (%)	Recall (%)	GFLOPs	Params
YOLOv5n	5.03	98.00	66.48	97.01	95.26	7.1	2,503,139
YOLOv6n	8.30	98.18	66.47	97.47	95.42	11.8	4,233,843
YOLOv8n	5.98	98.37	68.75	97.64	96.26	8.1	3,005,843
YOLOv10n	5.49	97.87	67.96	97.02	95.23	8.2	2,694,806
FasterNet-YOLOv8n	3.53	98.17	65.41	96.59	95.24	5.0	1,745,371
MobileNetV1-YOLOv8n	11.8	97.67	68.81	97.43	94.85	15.7	6,064,339
MobileNetV2-YOLOv8n	7.51	98.08	68.87	96.45	96.11	10.1	3,756,723
MobileNetV3-YOLOv8n	11.1	98.14	69.04	97.72	95.80	10.6	5,655,289
ShuffleNetV1-YOLOv8n	6.94	98.23	66.22	96.90	95.08	6.5	3,475,875
ShuffleNetV2-YOLOv8n	5.64	98.36	67.89	96.97	96.36	7.4	2,790,247
MobileNetV4-YOLOv8n-SPPF	4.45	98.56	69.43	97.66	95.04	6.2	2,183,451

The comparative results indicate that FasterNet-YOLOv8n achieves the lowest number of parameters and GFLOPs among all evaluated models. However, its mAP50–95 is only 65.41%, leading to noticeable missed detections and making it unsuitable for strawberry harvesting tasks that require high detection accuracy. In contrast, MobileNetV4-YOLOv8n-SPPF demonstrates a significant advantage in model size. Compared with YOLOv5n, YOLOv6n, YOLOv8n, MobileNetV1-YOLOv8n, MobileNetV2-YOLOv8n, MobileNetV3-YOLOv8n, ShuffleNetV1-YOLOv8n, and ShuffleNetV2-YOLOv8n, the model size is reduced by 0.58 MB, 3.85 MB, 1.53 MB, 1.04 MB, 7.30 MB, 3.06 MB, 6.65 MB, 2.49 MB, and 1.19 MB, respectively. Meanwhile, this model achieves the best performance in terms of mAP50 and mAP50–95, while maintaining comparable Precision and Recall to other models. In addition, its parameter count and computational cost are second only to FasterNet-YOLOv8n, while still remaining at a relatively low level. Considering both detection accuracy and model efficiency, MobileNetV4-YOLOv8n-SPPF is more suitable for deployment on edge devices for strawberry detection tasks.

The comparative results indicate that FasterNet-YOLOv8n achieves the lowest number of parameters and GFLOPs among all evaluated models. However, its mAP50–95 is only 65.41%, leading to noticeable missed detections and making it less suitable for strawberry harvesting tasks that require relatively high detection reliability. In contrast, MobileNetV4-YOLOv8n-SPPF demonstrates a clear advantage in model compactness. Compared with YOLOv5n, YOLOv6n, YOLOv8n, MobileNetV1-YOLOv8n, MobileNetV2-YOLOv8n, MobileNetV3-YOLOv8n, ShuffleNetV1-YOLOv8n, and ShuffleNetV2-YOLOv8n, the model size is reduced by 0.58 MB, 3.85 MB, 1.53 MB, 1.04 MB, 7.30 MB, 3.06 MB, 6.65 MB, 2.49 MB, and 1.19 MB, respectively.

Meanwhile, the proposed model achieves competitive performance in terms of mAP50 and mAP50–95, while maintaining comparable Precision and Recall to other lightweight configurations. Its parameter count and computational cost are second only to FasterNet-YOLOv8n, but remain at a relatively low level overall. This suggests that the proposed approach provides a balanced trade-off between model compactness and detection performance.

Considering both detection accuracy and model efficiency, MobileNetV4-YOLOv8n-SPPF is well-suited for deployment on edge devices for strawberry detection tasks, especially in scenarios where both real-time performance and reliable perception are required.

Compared with recent YOLO-based strawberry detection methods reported in the literature (e.g., SDNet [15], DSW-YOLO [27], and STRAW-YOLO [25]), the proposed model maintains a relatively low parameter scale while achieving a balanced trade-off between lightweight design and detection accuracy. This balance makes the method more suitable for deployment on resource-constrained platforms.

In addition, beyond the detection stage, the proposed approach integrates a weighted geometric centroid reconstruction method and IMU-based dynamic compensation, enabling more stable and reliable spatial localization for harvesting tasks. While existing studies mainly focus on improving detection performance or keypoint estimation, the proposed method further considers the requirements of downstream robotic operations, providing a more complete and task-oriented solution for practical strawberry harvesting scenarios.

3.1.4. Model Testing of Strawberry Picking Robots

The Triplet Attention mechanism introduces cross-dimensional feature interactions to jointly model channel-wise and spatial information, thereby enhancing the representation of salient regions. In scenarios where strawberry fruits are densely distributed and heavily occluded by leaves and branches, local features of targets are often interfered with by complex background information, leading to weakened responses or even missed detections. To address this issue, the introduction of Triplet Attention enables the model to generate attention weights dynamically based on the input feature distribution and to adjust feature responses across different spatial locations and channels. This process highlights informative regions while suppressing irrelevant background interference. As a result, the edge contours and fine-grained texture features of occluded fruits are better preserved, improving their discriminability. In addition, multi-directional feature interactions further enhance the consistency of feature representation, allowing the model to maintain stable detection performance for partially visible targets in complex environments.

As shown in Figure 12, the blue boxes indicate the predicted bounding boxes output by the models.The improved model demonstrates significantly better detection capabilities in complex scenarios such as intensive targets, occlusion, and backlight conditions compared to the original model. In dense target scenarios, the comparison between Figure 12a,d shows that the improved model can detect more strawberries, indicating its enhanced ability to distinguish adjacent targets. This is attributed to Triplet Attention, which strengthens feature interaction between channels and spatial dimensions, allowing the model to maintain high recognition accuracy even when dealing with targets that are closely spaced. In occlusion scenarios, as seen in Figure 12b,e, the original model fails to accurately identify some occluded strawberries, whereas the improved model detects more targets. This suggests that Triplet Attention enhances the model's ability to focus on local information, enabling it to make reasonable inferences and accurate judgments even when parts of the target are occluded by utilizing surrounding information. Additionally, in backlit environments, as illustrated in Figure 12c,f, due to the impact of lighting, the contrast in the target regions decreases, making it difficult for the original model to accurately detect all strawberries. However, the improved model performs better, detecting more targets, which demonstrates that Triplet Attention enhances the model's feature extraction ability in low-contrast areas, allowing it to maintain high detection performance even under significant changes in lighting. Therefore, the superior performance of the improved model in various complex environments primarily benefits from Triplet Attention's effective enhancement in feature strengthening, information interaction, and local feature extraction, making the model more capable of distinguishing dense targets, more adept at inferring occluded targets, and more robust in detection under dramatic lighting changes. To further validate the performance of the improved MobileNetV4 + Triplet Attention + YOLOv8n model in embedded systems, this study deploys different YOLOv8 models on Jetson Nano and Orange Pi 5 PRO, testing their FPS.

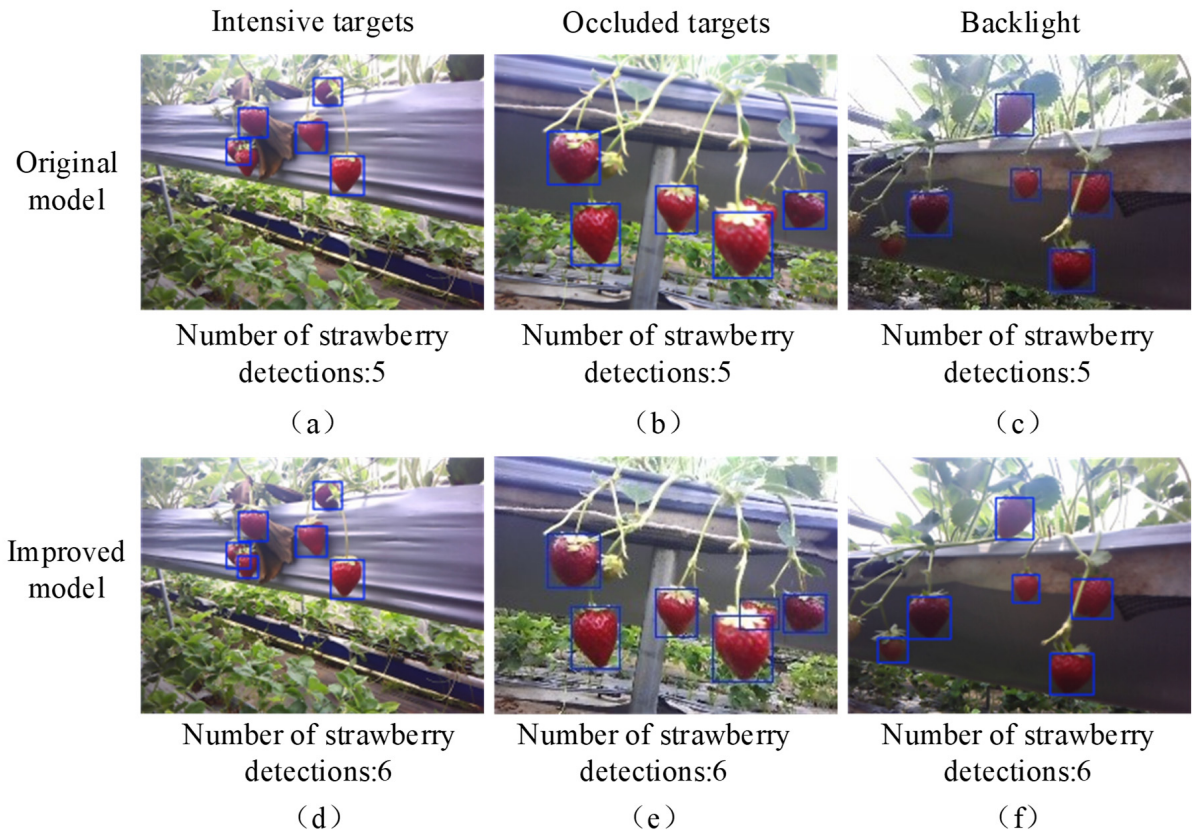


Figure 12. Detection results of the original YOLOv8n and improved YOLOv8n model in different environments. (**a**) The result of the original model processing intensive targets; (**b**) The result of the original model processing occluded targets; (**c**) The result of the original model processing the backlight scene targets; (**d**) The result of the improved model processing intensive targets; (**e**) The result of the improved model processing occluded targets; (**f**) The result of the improved model processing the backlight scene targets.

Table 3 shows the comparison results of inference speed for different lightweight models on two embedded platforms, Jetson Nano and Orange Pi. It can be seen that the original YOLOv8n achieved 15 FPS and 16 FPS on Jetson Nano and Orange Pi, respectively, serving as the baseline. Some alternative backbone designs, such as MobileNetV1-YOLOv8n and MobileNetV2-YOLOv8n, resulted in inference speeds of 13 FPS and 10 FPS, indicating that these configurations did not provide clear advantages in terms of real-time performance on the tested platforms. In contrast, the ShuffleNet and FasterNet series achieved speeds comparable to YOLOv8n, but without a significant improvement.

Table 3. FPS on Jetson Nano and Orange Pi 5 PRO.

Models	FPS in Jetson Nano	FPS in Orange Pi
YOLOv8n	15	16
FasterNet-YOLOv8n	24	18
MobileNetV1-YOLOv8n	13	10
MobileNetV2-YOLOv8n	13	10
MobileNetV3-YOLOv8n	16	11
ShuffleNetV1-YOLOv8n	23	16
ShuffleNetV2-YOLOv8n	25	19
Ours (MobileNetV4 + Triplet Attention + YOLOv8n)	30	22

In related studies, higher-performance embedded platforms (e.g., Jetson Xavier NX [17,27]) are often adopted to ensure real-time performance. Compared with these platforms, the Jetson Nano used in this study has more limited computational resources, making it a more challenging deployment condition.

Notably, the method proposed in this paper, Ours (MobileNetV4 + Triplet Attention + YOLOv8n), achieved 30 FPS and 22 FPS on Jetson Nano and Orange Pi, respectively, not only surpassing the original YOLOv8n in speed with an improvement of approximately 7% and 5%, but also maintaining reliable detection performance while ensuring real-time inference. This suggests that the proposed method provides a task-oriented balance between efficiency and accuracy under relatively constrained hardware conditions, making it suitable for practical harvesting scenarios.

3.2. Verification of 3D Positioning Accuracy and IMU-Assisted Compensation Effect

3.2.1. Single-Frame 3D Positioning Accuracy Test

To evaluate the effectiveness of the method proposed in this paper, we recorded 100 sets of strawberry images and depth maps under different occlusion levels in Yunchuang Farm, Panlong District, Kunming City, Yunnan Province. Subsequently, we used the improved YOLOv8n for inference to obtain the bounding boxes of strawberries in each image. In field tests, the occluders of strawberries can be roughly divided into five categories: fruit overlap occlusion, unripe fruit occlusion, calyx occlusion, leaf occlusion, and stem occlusion. According to the degree of occlusion, they are further divided into slight occlusion, moderate occlusion, and severe occlusion. In actual harvesting scenarios, slight and moderate occlusion under different occluder conditions are more common. Figure 13 shows some examples of strawberries with 3D centroid reconstruction in real-world scenarios, where the white boxes are the bounding boxes predicted by the model. In the results, the white points represent the center points of the bounding boxes, and the red points represent the coordinates after centroid reconstruction. In cases of slight occlusion, the results of centroid reconstruction almost coincide with the center points of the bounding boxes, indicating that when the depth is continuous inside the bounding box and there is no obvious interference, the performance of the two methods is close, as shown in Figure 13 for leaf occlusion and stem occlusion. When the occluder is exactly at the center of the bounding box, the positioning point can also be accurately selected on the strawberry.

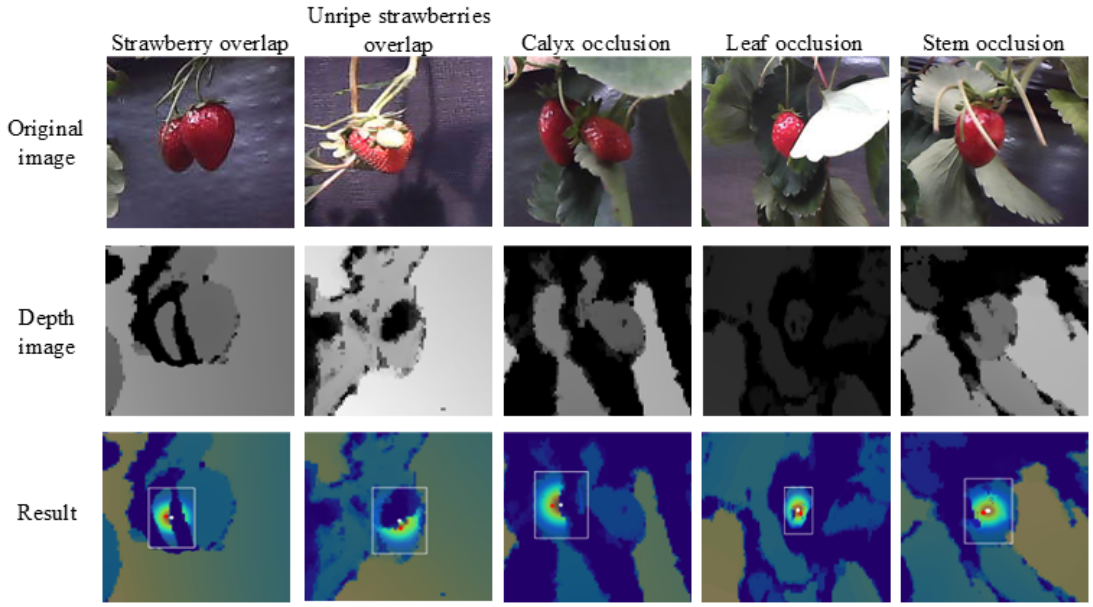


Figure 13. Pixel-weighted centroid reconstruction results under different occlusion conditions.

After testing, the 100 sets of photos collected on-site contained a total of 216 strawberries, of which 209 were successfully completed with 3D reconstruction. The remaining 7 strawberries, 5 failed to complete reconstruction due to the lack of effective depth information within the bounding box, and 1 was offset to the occluder due to large-scale occlusion, as shown in Figure 14. The failure of the former is mainly due to the strong natural light in the strawberry greenhouse interfering with the projection of structured light, making it impossible for the depth camera to correctly measure the distance to the target surface, resulting in large areas of "depth voids" (depth value of 0) in the depth map, ultimately leading to no usable depth data within the bounding box. The deviation of the latter occurs under extremely severe occlusion conditions, when the occluder occupies most of the area of the bounding box and is near the geometric center, the weighted centroid algorithm tends to assign higher spatial distribution weight and depth continuity weight to the occluder pixels during the calculation process, ultimately causing the reconstruction point to fall on the occluder rather than the fruit surface.

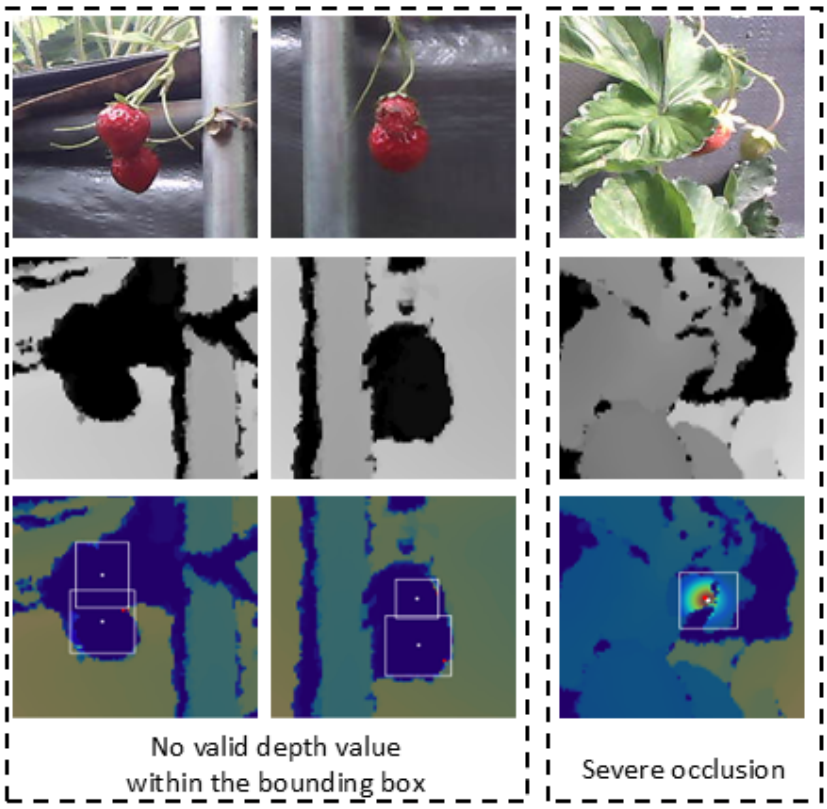


Figure 14. Examples of failed centroid reconstruction.

To further evaluate the performance of the proposed weighted geometric centroid reconstruction method, comparative experiments are conducted against two baseline approaches: the traditional center-point depth method and a region-of-interest (ROI) refinement method with median depth extraction. The effectiveness of the proposed method in improving the accuracy of fruit 3D localization is validated under varying occlusion conditions.

As illustrated in Figure 15, the experimental setup consists of a fixed-depth camera mounted on a stable platform, a strawberry holder with an adjustable position, and a calibrated reference base. A precision ruler is used to obtain the ground truth coordinates $(X_{true}, Y_{true}, Z_{true})$ in the camera coordinate system. By adjusting the relative positions of the strawberry and the camera, a total of nine experimental scenarios are constructed, covering four representative occlusion conditions: no occlusion (fully visible fruit), slight

occlusion (partially covered by leaves), moderate occlusion (partial edge occlusion), and severe occlusion (most of the fruit boundary occluded).

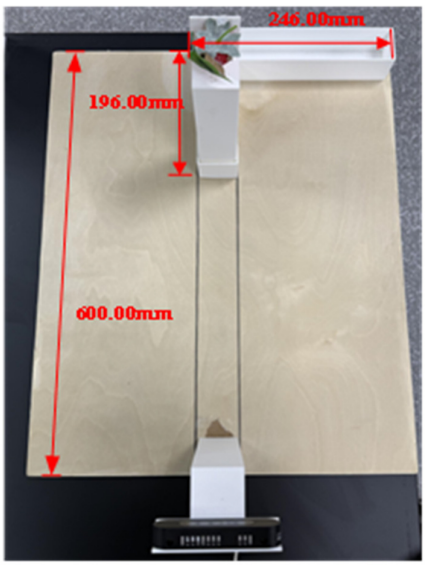


Figure 15. Experimental scene diagram.

For each scenario, 50 repeated measurements are performed, and the averaged results are used for evaluation. The performance of different methods is quantitatively assessed using the root mean square error (RMSE) as the primary metric, which reflects both accuracy and sensitivity to outliers. In addition, the standard deviation (STD) is reported to evaluate the stability of the reconstruction results.

The three methods compared in this experiment include (1) the conventional method based on the bounding box center point with corresponding depth value, (2) an ROI-based method that reduces the detection region and extracts the median depth to suppress noise, and (3) the proposed weighted geometric centroid reconstruction method, which integrates spatial weighting and depth consistency to achieve more accurate and robust 3D position estimation.

From the experimental results presented in Table 4, it can be observed that the proposed weighted centroid method consistently outperforms the traditional center-point depth method across different occlusion conditions, while also demonstrating clear advantages over the ROI-based median depth method, particularly under moderate and severe occlusion. To provide a more intuitive evaluation, the errors are further normalized with respect to the average strawberry diameter (4.5 cm) and reported as relative errors.

Under no-occlusion conditions, the depth distribution within the bounding box remains continuous and stable. As a result, all three methods achieve comparable performance, with only marginal differences. Specifically, the MAE of the weighted centroid method is 0.19 cm (4.22%), compared to 0.21 cm (4.67%) for the central point method and 0.18 cm (4.00%) for the ROI-based method, indicating that the benefit of advanced strategies is limited when depth information is reliable.

However, as occlusion increases, the performance gap between methods becomes more pronounced. The traditional center-point method exhibits significant degradation due to its reliance on a single pixel location. For instance, under slight occlusion, its MAE increases to 0.65 cm (14.44%), and further rises to 0.89 cm (19.78%) under moderate occlusion. In contrast, the ROI-based method improves robustness by suppressing background noise through regional filtering, achieving MAEs of 0.42 cm (9.33%) and 0.66 cm (14.67%) under slight and moderate occlusion, respectively.

Table 4. Positioning error and standard deviation of center point method and weighted centroid reconstruction method.

Occlusion Conditions	Methods	MAE (cm)	STD (cm)	RMSE (cm)	Relative Error (%)
No occlusion	Central point method	0.21	±0.11	0.23	4.67
	ROI-Shrink + Median Depth	0.18	±0.10	0.21	4.00
	Weighted centroid method	0.19	±0.11	0.22	4.22
slight occlusion	Central point method	0.65	±0.25	0.71	14.44
	ROI-Shrink + Median Depth	0.42	±0.18	0.46	9.33
	Weighted centroid method	0.33	±0.15	0.40	7.33
Moderate occlusion	Central point method	0.89	±0.22	0.92	19.78
	ROI-Shrink + Median Depth	0.66	±0.21	0.70	14.67
	Weighted centroid method	0.54	±0.18	0.62	12.00
Severe occlusion	Central point method	0.93	±0.39	1.08	20.67
	ROI-Shrink + Median Depth	0.88	±0.30	0.98	19.56
	Weighted centroid method	0.73	±0.23	0.81	16.22

The proposed weighted centroid method demonstrates superior performance by incorporating spatial weighting and depth consistency. It achieves MAEs of 0.33 cm (7.33%), 0.54 cm (12.00%), and 0.73 cm (16.22%) under slight, moderate, and severe occlusion conditions, respectively, with corresponding RMSE values of 0.40 cm, 0.62 cm, and 0.81 cm. Compared to the baseline methods, it maintains lower absolute and relative errors, as well as reduced variance, indicating improved robustness and stability under complex occlusion scenarios.

To further determine the optimal parameter combination of the weighted 3D centroid reconstruction algorithm and validate its stability, a full-parameter grid search was conducted on the spatial scaling factor σ and the depth bias adjustment coefficient κ. A total of 50 strawberry samples with varying levels of occlusion and fruit sizes were selected as the test set. The geometric center of each strawberry was manually annotated in the camera coordinate system using a high-precision ruler as the ground truth (GT).

During the experiments, the search range of σ was set to $[0.05, 0.80]$ with a step size of 0.05, while κ was varied within $[5, 50]$ mm with a step size of 5 mm. The root mean square error (RMSE) was adopted as the evaluation metric to comprehensively assess localization accuracy and sensitivity to outliers.

As illustrated in Figure 16, the RMSE distribution forms a smooth and continuous basin-shaped surface in the σ–κ parameter space, with a clearly identifiable global minimum(the value represented by the blue dot). The error surface gradually decreases from the outer regions toward the central valley and then increases again, indicating a well-defined optimal region rather than isolated local minima. Specifically, for a fixed κ, the RMSE decreases initially and then increases as σ grows, forming a convex trend along the σ direction.

The lowest error is observed around $\sigma \approx 0.25$, which corresponds to approximately one-quarter of the bounding box width. This suggests that the selected spatial weighting effectively balances the contribution of central pixels and the suppression of noisy depth measurements from occluded boundaries. When σ is too small ($\sigma < 0.15$), the weighting becomes overly concentrated, making the estimation sensitive to local depth noise. In contrast, when σ is too large ($\sigma > 0.50$), the inclusion of peripheral pixels introduces interference from surrounding leaves and stems, resulting in centroid bias.

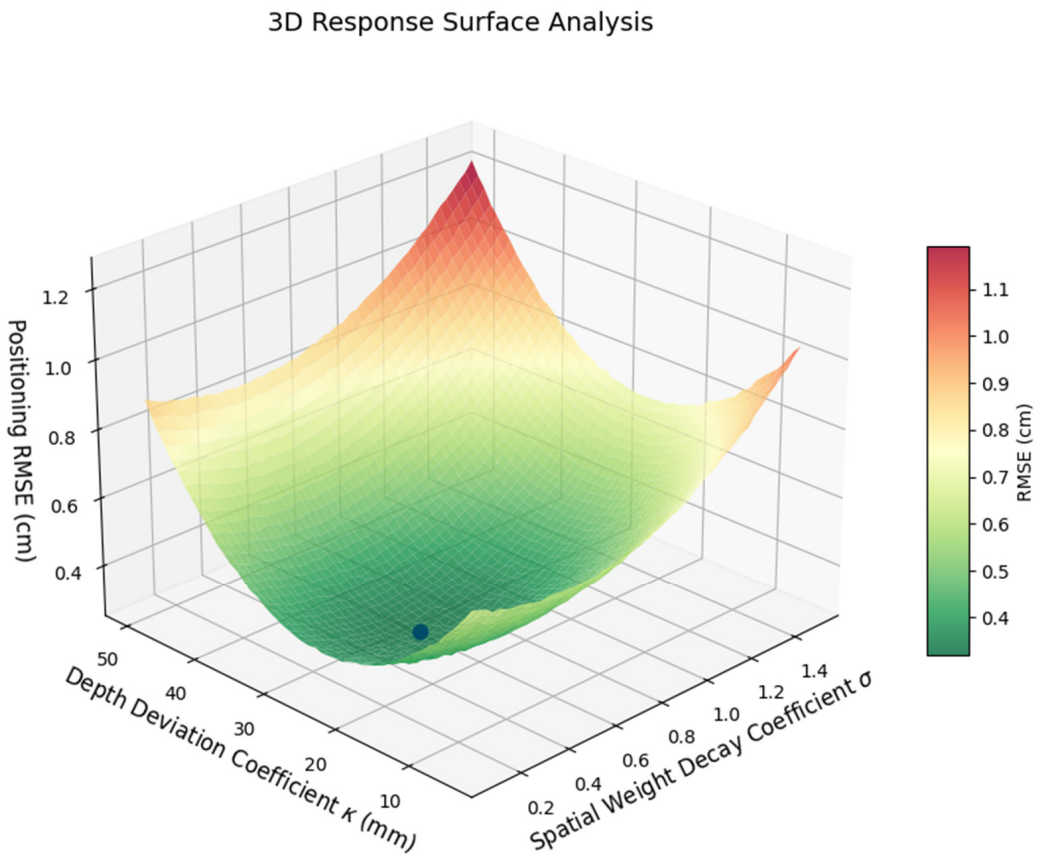


Figure 16. RMSE distribution over the parameter space of σ and κ.

Along the κ dimension, the RMSE surface shows a clear minimum near $\kappa \approx 25$ mm, where the global minimum RMSE (approximately 0.32 cm) is achieved. When $\kappa \leq 15$ mm, the depth filtering window is too narrow, leading to the loss of valid surface points and insufficient geometric representation. Conversely, when $\kappa \geq 40$ mm, the window becomes excessively large, causing background depth values and neighboring structures to be incorporated, which significantly degrades localization accuracy.

Furthermore, the relatively flat and wide valley region around the optimal point ($\sigma \approx 0.25$, $\kappa \approx 25$ mm) indicates low sensitivity to parameter perturbations. This characteristic demonstrates that the proposed method maintains stable performance even under moderate variations in fruit size and occlusion conditions. Such a smooth error landscape confirms the robustness and practical applicability of the weighted centroid reconstruction method in unstructured orchard environments.

Nevertheless, it can be observed that the accuracy of all methods degrades under severe occlusion. Although the proposed method still achieves the best performance (MAE = 0.73 cm, STD = $\pm$0.23 cm), the error increase is unavoidable. This is because, in such scenarios, the proportion of valid depth pixels within the detection region is significantly reduced, and the remaining depth information may be heavily contaminated by occluders. As a result, the reliability of the weighted centroid estimation is affected, leading to deviations from the true fruit position. Representative examples of such cases are illustrated in Figure 17.

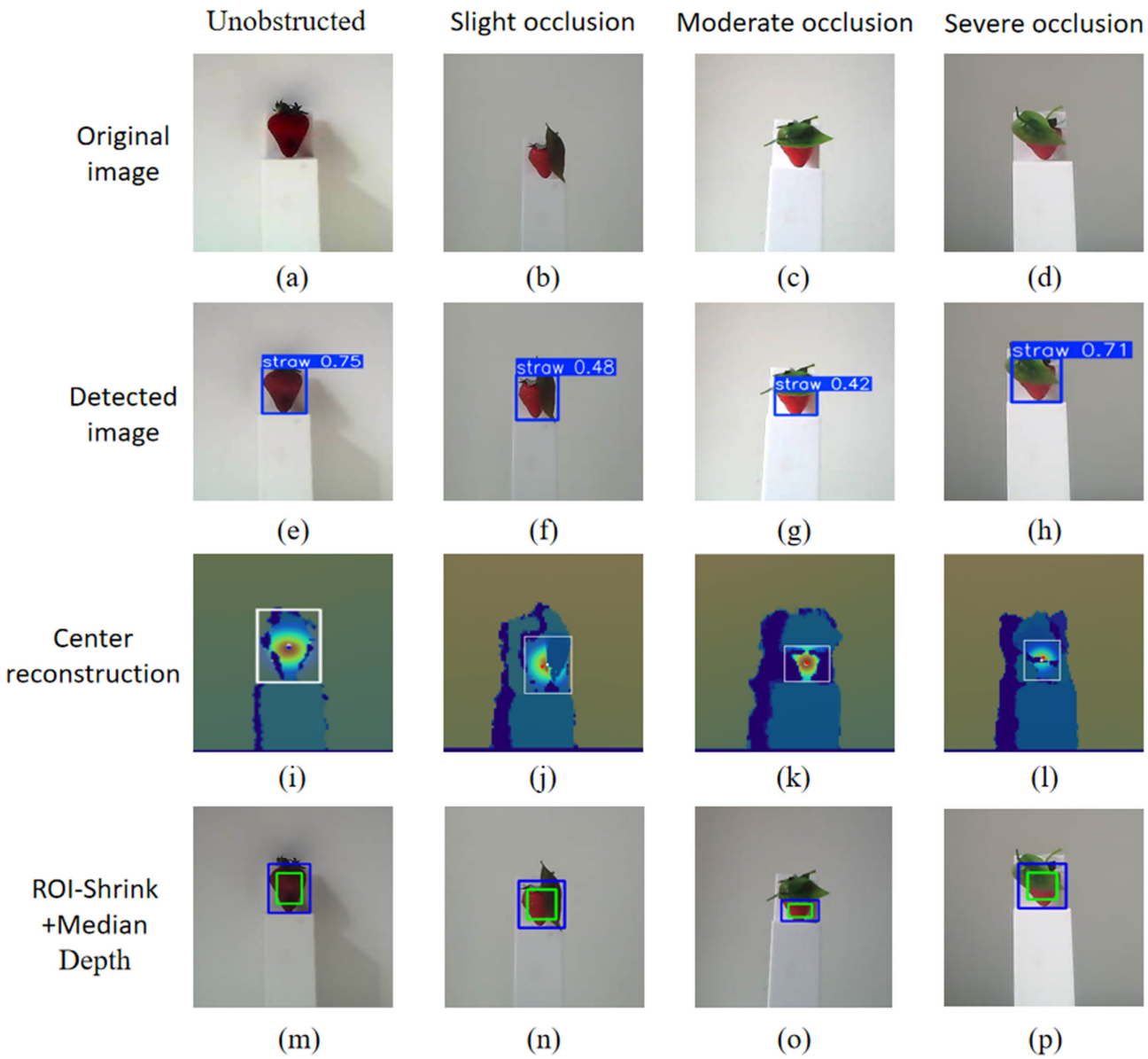


Figure 17. Comparison of centroid reconstruction effects under occlusion and non-occlusion.

As illustrated in Figure 17, qualitative comparisons of different reconstruction strategies under varying occlusion conditions are presented. The first row (Figure 17a–d) shows the original RGB images under four conditions: no occlusion, slight occlusion, moderate occlusion, and severe occlusion. The second row (Figure 17e–h) presents the corresponding detection results obtained by the improved YOLOv8n model. It can be observed that, under occlusion, the predicted bounding boxes inevitably include partial occluders, which introduces challenges for accurate depth estimation. The third row (Figure 17i–l) illustrates the results of the proposed weighted centroid reconstruction method in the form of spatial weight distributions. It can be seen that pixels located on the visible fruit surface are assigned higher weights, forming a concentrated high-response region. The final centroid is obtained through a weighted aggregation over the entire distribution rather than selecting a single maximum point, which enables the method to better capture the true geometric center of the fruit. Even under slight and moderate occlusion (Figure 17j,k), the high-weight region remains primarily distributed over the exposed fruit area, demonstrating strong robustness against occlusion interference. The fourth row (Figure 17m–p) shows the results of the ROI-shrink combined with median depth extraction, where the blue boxes are pre-

diction boxes and the green ones are ROIs based on the contraction of prediction boxes. By reducing the effective region and applying median filtering, this method alleviates part of the background noise and improves robustness compared to simple center-point estimation. However, due to the absence of spatial weighting, the reconstructed results are still influenced by the distribution of remaining pixels. In particular, under severe occlusion (Figure 17p), the estimated position tends to deviate from the actual fruit center.

Overall, the comparison indicates that the proposed weighted centroid method can effectively suppress the influence of occluders by emphasizing reliable pixels and reducing the contribution of noisy or irrelevant regions. Nevertheless, under extreme conditions where the visible portion of the fruit is severely limited, the available valid depth information becomes insufficient, which may still lead to deviations in the reconstructed centroid.

In addition, it should be noted that the performance of the system is also affected by the characteristics of the depth sensor. For example, structured-light depth cameras may experience reduced measurement reliability under strong natural illumination, potentially introducing additional noise into depth estimation. Furthermore, the current weighting strategy is primarily based on empirical design and lacks full adaptability. Future work could explore multi-sensor fusion approaches, such as integrating depth cameras with LiDAR, and incorporating lightweight semantic segmentation techniques to further improve the discrimination between fruits and occluding objects.

3.2.2. IMU Compensation Experiment Under Dynamic Interference

To verify the effectiveness of the proposed IMU-assisted dynamic compensation and multi-frame fusion method under dynamic interference conditions, this paper designs and constructs a controllable simulation vibration test platform, as shown in Figure 18. The platform is installed 200 mm away from the strawberry experimental bench, used to generate multi-directional, random disturbances, thereby simulating dynamic interferences such as the vibration of mechanical arms during orchard operations, platform displacement, or uneven ground. The vibration device consists of two servo motors and an ESP32 control board. The servo motors are used to generate disturbances in the horizontal and pitch directions, while the ESP32 drives the servos via PWM signals to execute pseudo-random control sequences, causing the camera to exhibit irregular attitude jitter in multiple directions. By adjusting the amplitude and frequency of servo deflection, different intensities of disturbances can be flexibly set, thus achieving controllable simulation of the experimental environment.

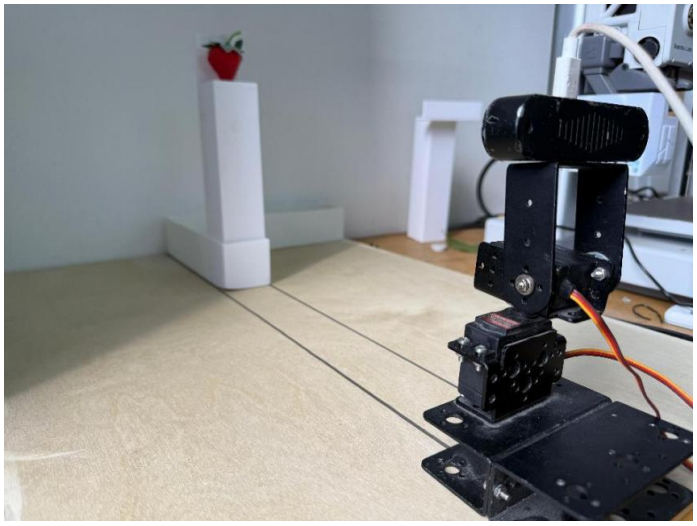


Figure 18. Simulation vibration test setup.

To make the simulated vibration closer to real conditions, this paper recorded the original IMU data during the actual operation of the harvesting robot, including tri-axial acceleration $a_t = [a_x, a_y, a_z]$ and tri-axial angular velocity $\omega_t = [\omega_x, \omega_y, \omega_z]$, and defined a quantitative index for disturbance intensity δt according to Formula (20) in Section 2.6.

Based on the statistical distribution, the experiment divided the disturbances into slight jitter ($\delta_t < \eta_1$), moderate jitter ($\eta_1 \leq \delta_t < \eta_2$) and severe jitter ($\delta_t \geq \eta_2$). Among them, $\eta_1 = 0.8$, $\eta_2 = 2.0$. Experiments were conducted on the simulation platform for each corresponding disturbance level. Throughout the experimental process, the camera captured RGB images and depth images at 30 fps, and the IMU recorded acceleration and angular velocity at 200 Hz. All data were timestamped and aligned through interpolation to ensure consistency between visual information and inertial information. Under stable conditions, the true coordinates of the strawberries were taken as (5, 5, 20). Multiple independent tests were executed under different disturbance levels, with 100 positioning points taken each time to ensure the statistical reliability of the results.

Throughout the experimental process, the camera captured RGB images and depth images at 30 fps, and the IMU recorded acceleration and angular velocity at 200 Hz. To address the sampling frequency mismatch, we implemented a linear interpolation method for timestamp alignment. Specifically, for each camera frame at t_{cam}, we identified the two closest IMU readings at t_k and t_{k+1} ($t_k \leq t_{cam} < t_{k+1}$) and calculated the synchronized inertial state via

$$D_{sync} = D_k + (D_{k+1} - D_k) \cdot \frac{t_{cam} - t_k}{t_{k+1} - t_k}$$

Since the IMU sampling interval is 5 ms, the maximum theoretical synchronization jitter is significantly reduced through interpolation.

In terms of method comparison, four schemes are designed for evaluation. The first is the uncompensated method, which directly uses single-frame 3D coordinates obtained from detection and depth reconstruction. The second is the baseline method, where a sliding window with equal weights is applied to average the 3D coordinates of consecutive frames, aiming to suppress short-term jitter. The third method introduces IMU-assisted compensation, in which the rotation matrix estimated from IMU measurements is used to correct single-frame coordinates, followed by a dynamic adaptive-weight sliding window fusion to improve robustness under motion disturbances. The fourth method corresponds to the proposed approach, where an Extended Kalman Filter (EKF) is further applied to the fused coordinates to model temporal dynamics and refine the estimation results. This hierarchical experimental design enables a comprehensive evaluation of the contributions of sliding window smoothing, IMU-based compensation, and EKF-based temporal refinement to the overall localization performance.

To comprehensively evaluate the effectiveness of the methods, the experiment used 3D positioning mean absolute error (MAE), standard deviation (STD), and root mean square error (RMSE). The experimental results are shown in Table 5.

Table 5 presents the error and stability performance of four methods under different vibration levels. To provide a scale-invariant evaluation, the errors are additionally normalized with respect to the average strawberry diameter (4.5 cm) and reported as relative errors. The results demonstrate that the proposed IMU-assisted strategies significantly improve localization accuracy and robustness compared with the baseline methods, and that the integration of EKF further enhances performance.

Table 5. Performance of Error and Stable Frame Ratio under Different Vibration Levels.

Vibration Levels	Methods	MAE (cm)	STD (cm)	RMSE (cm)	Relative Error (%)
$\delta_t < \eta_1$	No compensation	1.21	±0.52	1.32	26.89
	Sliding window	0.68	±0.34	0.76	15.11
	IMU + weighted sliding window	0.46	±0.24	0.52	10.22
	IMU + weighted sliding window + EKF	0.28	±0.16	0.32	6.22
$\eta_1 \leq \delta_t < \eta_2$	No compensation	2.23	±0.91	2.34	49.56
	Sliding window	1.35	±0.62	1.48	30.00
	IMU + weighted sliding window	0.68	±0.36	0.77	15.11
	IMU + weighted sliding window + EKF	0.40	±0.20	0.45	8.89
$\delta_t \geq \eta_2$	No compensation	2.45	±1.06	2.55	54.44
	Sliding window	1.58	±0.95	1.85	35.11
	IMU + weighted sliding window	0.92	±0.52	1.05	20.44
	IMU + weighted sliding window + EKF	0.51	±0.30	0.57	11.33

Under slight vibration conditions ($\delta_t < \eta_1$), all methods maintain relatively good performance due to limited dynamic disturbance, while noticeable differences can still be observed. The uncompensated method exhibits an MAE of 1.21 cm (26.89%) and an RMSE of 1.32 cm (29.33%), indicating the presence of inherent jitter even under low disturbance. Sliding window smoothing reduces short-term fluctuations, lowering the MAE to 0.68 cm (15.11%) and the RMSE to 0.76 cm (16.89%). The IMU-assisted weighted sliding window method further improves the results, achieving an MAE of 0.46 cm (10.22%) and an RMSE of 0.52 cm (11.56%), demonstrating its ability to compensate for minor motion disturbances. With the introduction of EKF, the error is further reduced to an MAE of 0.28 cm (6.22%) and an RMSE of 0.32 cm (7.11%), indicating improved temporal consistency and filtering effectiveness.

Under moderate vibration conditions ($\eta_1 \leq \delta_t < \eta_2$), the differences among the methods become more pronounced. The uncompensated method shows significant degradation, with an MAE of 2.23 cm (49.56%) and an RMSE of 2.34 cm (52.00%), reflecting poor robustness under dynamic disturbance. Sliding window smoothing provides limited improvement, reducing the MAE to 1.35 cm (30.00%) and the RMSE to 1.48 cm (32.89%), but still suffers from accumulated motion error. In contrast, the IMU-assisted method maintains strong robustness, achieving an MAE of 0.68 cm (15.11%) and an RMSE of 0.77 cm (17.11%), effectively suppressing motion-induced deviations. The EKF-enhanced method further improves accuracy, reducing the MAE to 0.40 cm (8.89%) and the RMSE to 0.45 cm (10.00%), demonstrating its advantage in modeling dynamic system behavior and reducing cumulative errors.

Under severe vibration conditions ($\delta_t \geq \eta_2$), the performance gap becomes even more significant. The uncompensated method exhibits large errors (MAE = 2.45 cm (54.44%), RMSE = 2.55 cm (56.67%)), indicating poor reliability in highly disturbed environments. Although sliding window smoothing alleviates fluctuations to some extent (MAE = 1.58 cm (35.11%), RMSE = 1.85 cm (41.11%)), its compensation capability remains limited due to the lack of motion awareness. The IMU-assisted weighted sliding window method significantly improves performance, achieving an MAE of 0.92 cm (20.44%) and an RMSE of 1.05 cm (23.33%), demonstrating its effectiveness in compensating for strong disturbances. Furthermore, the proposed IMU + EKF method achieves the best results, with the MAE reduced to 0.51 cm (11.33%) and the RMSE to 0.57 cm (12.67%), highlighting its superior robustness and stability under severe vibration conditions.

Overall, the results indicate that while sliding window smoothing can effectively reduce high-frequency noise, it cannot fundamentally eliminate motion-induced errors. The IMU-assisted method addresses this limitation by incorporating motion compensation, and the integration of EKF further enhances temporal consistency by explicitly modeling system dynamics. Consequently, the combined IMU + EKF framework achieves the most accurate and stable localization performance across all vibration conditions.

As illustrated in the 3D point cloud distributions in Figure 19, the localization performance of four methods under different vibration levels (severe, moderate, and slight jitter) can be intuitively compared. The green points denote the ground truth position (5,5,20), while the red points represent the uncompensated results, which exhibit significant dispersion, especially under severe jitter conditions (Figure 19a,e,i). After applying sliding window smoothing (Figure 19b,f,j), the spatial distribution of the points becomes more concentrated, indicating that short-term fluctuations are effectively suppressed. However, noticeable deviations from the ground truth still remain due to uncorrected motion-induced errors. With the introduction of IMU-assisted compensation and adaptive weighted sliding window fusion (Figure 19c,g,k), the point cloud shows a significantly tighter clustering around the true position. This demonstrates that IMU-based pose correction effectively reduces systematic errors caused by platform motion, while the adaptive weighting mechanism further suppresses unreliable observations. Finally, as shown in Figure 19d,h,l, the proposed method incorporating EKF-based refinement achieves the most compact and stable distribution. By explicitly modeling temporal dynamics, the EKF further reduces residual fluctuations and enhances the continuity of the localization results across frames. A horizontal comparison across different vibration levels reveals that the performance gap between methods becomes more pronounced as motion intensity increases. Although the IMU-assisted adaptive fusion method significantly improves positioning stability, occasional outliers may still occur under severe vibration conditions (Figure 19c). This phenomenon is mainly attributed to the amplification of high-frequency noise and the accumulation of integration drift in IMU measurements under rapid motion, as well as transient inconsistencies between visual and inertial observations. In contrast, the proposed EKF-enhanced method effectively mitigates these residual anomalies by incorporating temporal consistency constraints, resulting in superior robustness and accuracy under all tested conditions.

3.3. Picking Experiment

To verify the effectiveness of the proposed method in an actual robotic system, this study designed and conducted a picking experiment at Yun Chuang Farm in Panlong District, Kunming City, Yunnan Province. The experiment was carried out under clear weather conditions with sufficient lighting. The farm uses greenhouse cultivation to optimize the growth environment of strawberries and effectively control temperature and humidity. As shown in Figure 20, the planting distance between adjacent rows of strawberries is approximately 0.75 m. During the strawberry ripening stage, each plant averages about 5–8 mature strawberries.

As shown in Table 6, after 200 field picking experiments, 174 successful picking attempts were recorded, resulting in an average picking success rate of 87%. This indicates that the proposed system can achieve high picking efficiency in a real orchard environment, but it is still affected by environmental complexities, such as changes in lighting, leaf and branch occlusion, and uncertainties in robotic arm movement, all of which may lead to failures or missed picks.

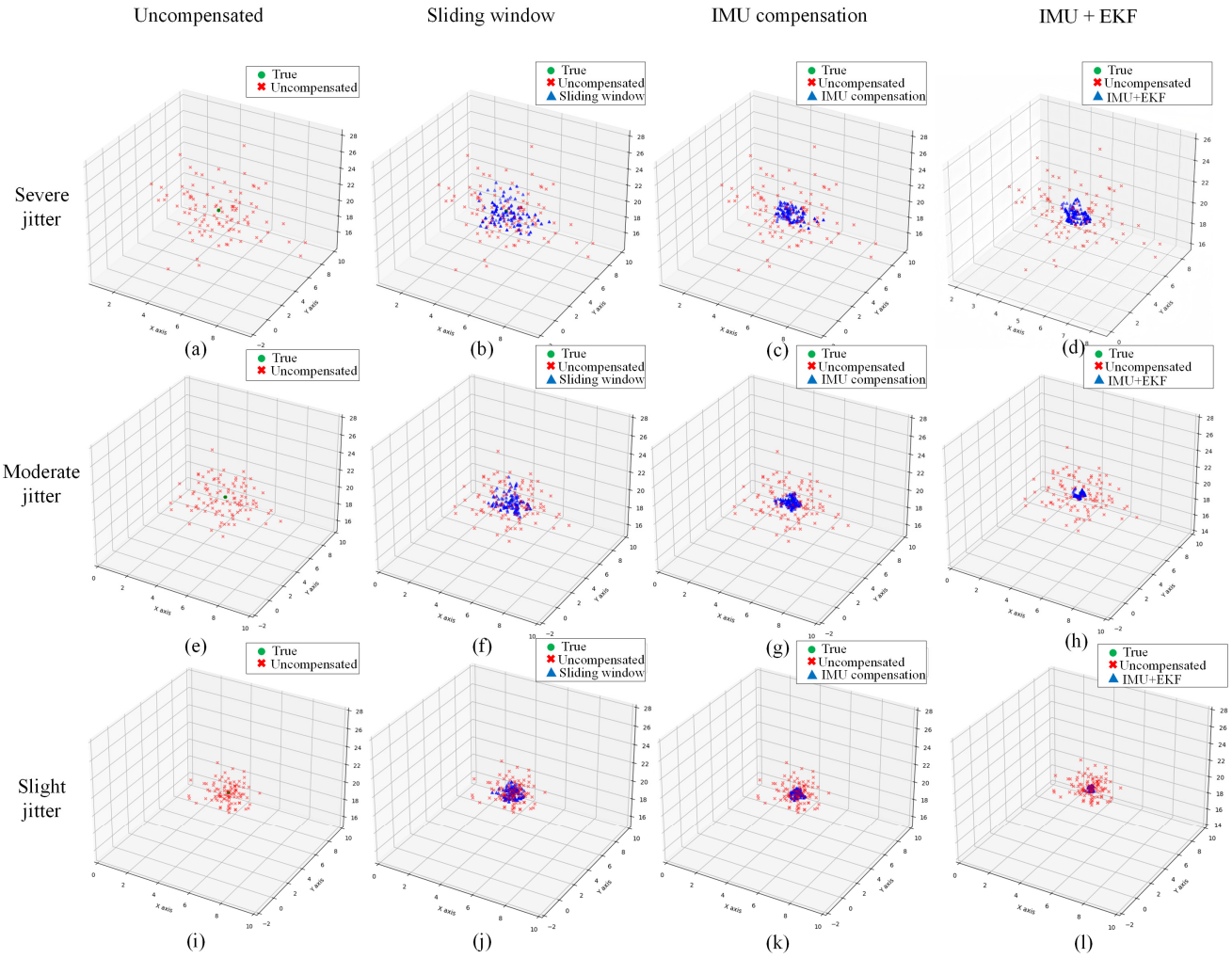


Figure 19. Three-axis curves and 3D point cloud maps of dynamic interference compensation for different methods.



Figure 20. Experimental site at Yun Chuang Farm, Panlong District, Kunming City.

Table 6. Results of Field Harvesting Experiment.

Number of Experiments	Number of Successful Picks	Number of Picking Failures	Number of Missed Picking	Picking Success Rate
200	174	16	10	87%

After applying the improved YOLOv8n object detection algorithm combined with pixel-weighted centroid reconstruction and visual-IMU fusion methods to the picking robot in this study, it was considered that it is difficult to conduct multiple picking experiments in actual sites, as shown in the Figure 21. The scene conditions set in the laboratory are all standardized and recorded, possessing complete reproducibility, allowing for multiple picks of the same fruit, effectively avoiding the impact of randomness on experimental results. Therefore, this study constructed a simple strawberry planting environment in the laboratory and conducted multiple picking experiments to enhance reliability. The laboratory operation scenario is shown in Figure 22, and the experimental results are presented in Table 7.



Figure 21. Actual operation scenario.

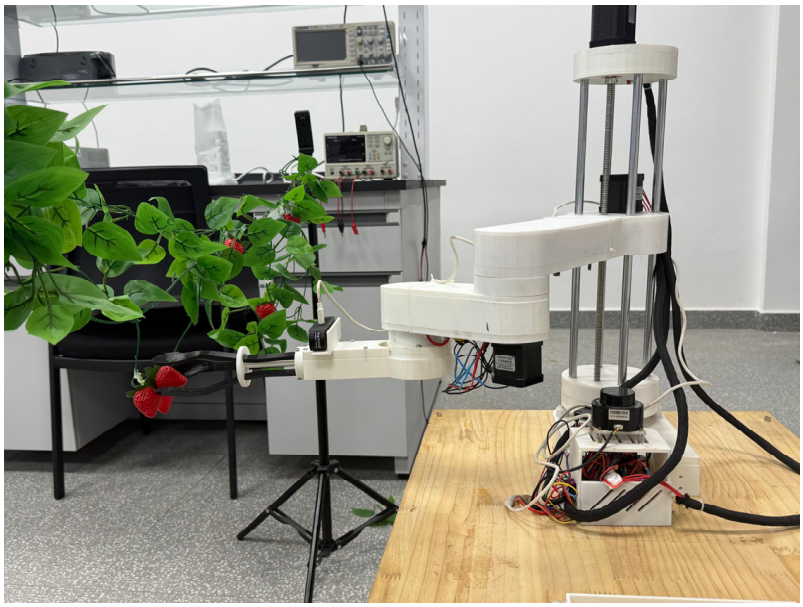


Figure 22. Laboratory work scenario diagram.

Table 7. Results of laboratory simulation harvesting experiment.

Improved Detector	Weighted Centroid	IMU Compensation	EKF	MAE (cm)	STD (cm)	RMSE (cm)	Relative Error (%)	Success Rate (%)
				2.36	±1.02	2.56	52.44	78.5
√				1.95	±0.88	2.14	43.33	83.2
√	√			1.18	±0.52	1.29	26.22	88.7
√	√	√		0.76	±0.45	0.88	16.89	90.3
√	√	√	√	0.51	±0.30	0.57	11.33	92.1

After integrating the improved YOLOv8n detector, the pixel-weighted centroid depth reconstruction method, and the visual–IMU fusion strategy with EKF into the picking robot system, systematic validation experiments were conducted. Considering the difficulty of performing large-scale repeated trials in real orchard environments, a controlled laboratory setup was constructed to ensure reproducibility. The laboratory scene was standardized with consistent lighting and background conditions, enabling repeated picking of the same target and effectively reducing randomness. The experimental setup is shown in Figure 22, and the quantitative results, including the system-level ablation study, are summarized in Table 7.

As presented in Table 7, the ablation results clearly demonstrate the individual and cumulative contributions of each module, where √ indicates that the corresponding module or method is applied. Starting from the baseline system without any enhancement, the localization error is relatively large (MAE = 2.36 cm, RMSE = 2.56 cm), and the picking success rate is limited to 78.5%. After introducing the improved detector, the success rate increases to 83.2%, indicating that more accurate fruit detection directly improves grasping reliability. When the pixel-weighted centroid depth reconstruction is further incorporated, the localization accuracy is significantly improved (MAE reduced to 1.18 cm), and the success rate rises to 88.7%, demonstrating the effectiveness of the proposed depth estimation strategy.

With the addition of IMU-assisted compensation, the system becomes more robust to dynamic disturbances, reducing motion-induced errors and further improving the success rate to 90.3%. Finally, by integrating EKF for temporal filtering and state estimation, the system achieves the best performance, with the localization error reduced to MAE = 0.51 cm and RMSE = 0.57 cm, while the picking success rate reaches 92.1%. This demonstrates that EKF effectively enhances temporal consistency and suppresses noise accumulation.

By comparing laboratory and field experiments, it can be observed that the system still achieves a picking success rate of 87% in complex orchard environments, demonstrating strong practical applicability. The higher success rate of 92.1% in the laboratory indicates that the proposed method has further performance potential under ideal conditions. The performance gap between the two scenarios suggests that environmental factors such as occlusion, illumination variation, and background complexity remain the primary challenges. Future work will focus on enhancing occlusion handling and improving illumination robustness to further narrow this gap and improve system stability in real-world applications.

4. Conclusions

The accurate localization and stable recognition of fruits are crucial for the efficient operation of picking robots. Although existing methods can achieve high accuracy under controlled conditions, they still face significant challenges in real orchard environments, including lighting fluctuations, fruit occlusion, and platform vibration. To address these limitations, this study proposes a comprehensive solution that integrates lightweight visual

detection, weighted 3D centroid reconstruction, and IMU-assisted dynamic compensation, while further improving overall operational efficiency through a priority picking strategy. These techniques are particularly valuable for industrially cultivated crops such as greenhouse-grown strawberries, where high-density planting and continuous production cycles demand precise, stable, and autonomous harvesting for sustainable industrial agriculture. The main conclusions are summarized below:

1. Fruit detection and recognition: A lightweight detection model based on an improved YOLOv8n is proposed, integrating the MobileNetV4 backbone and Triplet Attention mechanism. Compared to the original model, the parameter count is reduced by 25%, and mAP50–90 is improved by 1.201%. This significantly reduces the model's parameter count while maintaining high detection accuracy, enhancing real-time performance on embedded and edge devices.
2. 3D Coordinate Fusion: A weighted centroid reconstruction method was designed, integrating local pixel features of depth maps with spatial consistency. Experimental results show that under light and moderate occlusion conditions, the average localization error is controlled within 0.89 cm, with a standard deviation not exceeding $\pm$0.2 cm, demonstrating good robustness and repeatability. Even under severe occlusion, high reconstruction accuracy is maintained, reducing errors by approximately 12–20% compared to traditional methods. It effectively mitigates the impact of single-pixel depth noise, improving the stability and accuracy of fruit spatial position estimation.
3. IMU-Assisted Dynamic Compensation with EKF: To address localization errors caused by arm and platform vibrations in orchard environments, an integrated compensation framework combining IMU-based rotation matrix correction, a dynamically weighted sliding window, and Extended Kalman Filter (EKF) optimization is proposed. Specifically, IMU attitude estimation is utilized to compensate for motion-induced geometric deviations, while the weighted sliding window adaptively suppresses short-term noise. Furthermore, the EKF introduces temporal state estimation and dynamic modeling, enabling more consistent and robust localization results. Experimental results demonstrate that the proposed method significantly improves localization accuracy across all vibration levels. Under slight vibration conditions, the mean absolute error (MAE) is reduced to 0.28 cm with an RMSE of 0.32 cm. Under moderate and severe vibration conditions, the MAE remains at 0.40 cm and 0.51 cm, respectively, indicating strong robustness against dynamic disturbances. Compared with uncompensated and conventional smoothing methods, the proposed approach not only suppresses high-frequency noise but also effectively compensates for motion-induced errors and mitigates temporal drift. Even under severe vibration conditions, the localization error is maintained within approximately 0.6 cm, demonstrating its capability for accurate and stable real-time estimation of fruit spatial coordinates.
4. Multiple comparative experiments were conducted in Kunming City, Panlong District, Yunnan Province, at Yunchuang Farm and laboratory settings. In real-world environments, the success rate of harvesting reached 87%, while in laboratory environments, it was 92.1%. Experimental results indicate that the proposed visual-IMU fusion localization method achieves stable and reliable fruit localization under complex lighting, occlusion, and dynamic interference conditions, providing feasible technical support for the efficient application of orchard harvesting robots in real environments.

Overall, although this study focuses on greenhouse-grown strawberries, the proposed localization framework is not limited to this specific crop. Benefiting from its generalizable vision–inertial fusion architecture and strong robustness to occlusion, depth noise, and dynamic disturbances, the method has the potential to be extended to other industrially

cultivated crops—such as greenhouse berries, tomatoes, peppers, and various high-value horticultural plants. These crops often share similar characteristics, including dense canopy structures, frequent occlusion, and the need for continuous, stable, and automated harvesting. Future work will explore adapting the proposed approach to broader industrial cropping systems and integrating additional sensing modalities to further enhance its generalization and applicability in large-scale smart agriculture.

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

Funding: This work was supported by the Agricultural Joint Project of Yunnan Province [grant number 202301BD070001-127] and the Yunnan International Joint Laboratory of Natural Rubber Intelligent Monitor and Digital Applications [grant number 202403AP140001].

Data Availability Statement: The data presented in this study are available on request from the corresponding author. The data are not publicly available due to ongoing research use.

Acknowledgments: We sincerely thank Yunchuang Farm in Kunming, Yunnan, for their assistance in data set collection.

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

References

1. Bai, Y.; Zhang, B.; Xu, N.; Zhou, J.; Shi, J.; Diao, Z. Vision-based navigation and guidance for agricultural autonomous vehicles and robots: A review. *Comput. Electron. Agric.* **2022**, *205*, 107584. [CrossRef]
2. Eigenbrod, C.; Gruda, N. Urban vegetable for food security in cities: A review. *Agron. Sustain. Dev.* **2015**, *35*, 483–498. [CrossRef]
3. Xiang, J.; Wang, L.; Li, L.; Lai, K.-H.; Cai, W. Classification-design-optimization integrated picking robots: A review. *J. Intell. Manuf.* **2023**, *35*, 2979–3002. [CrossRef]
4. Wang, Z.; Xun, Y.; Wang, Y.; Yang, Q. Review of smart robots for fruit and vegetable picking in agriculture. *Int. J. Agric. Biol. Eng.* **2022**, *15*, 33–54. [CrossRef]
5. Li, Z.; Yuan, X.; Yang, Z. Design, simulation, and experiment for the end effector of a spherical fruit picking robot. *Int. J. Adv. Robot. Syst.* **2023**, *20*, 17298806231213442. [CrossRef]
6. Sibhatu, K.T.; Krishna, V.V.; Qaim, M. Production diversity and dietary diversity in smallholder farm households. *Proc. Natl. Acad. Sci. USA* **2015**, *112*, 10657–10662. [CrossRef]
7. Vougioukas, S.G. Agricultural Robotics. In *Annual Review of Control, Robotics, and Autonomous Systems*; Leonard, N.E., Ed.; Annual Reviews: Palo Alto, CA, USA, 2019; Volume 2, pp. 365–392.
8. Ghazal, S.; Munir, A.; Qureshi, W.S. Computer vision in smart agriculture and precision farming: Techniques and applications. *Artif. Intell. Agric.* **2024**, *13*, 64–83. [CrossRef]
9. Pandey, D.K.; Mishra, R. Towards sustainable agriculture: Harnessing AI for global food security. *Artif. Intell. Agric.* **2024**, *12*, 72–84. [CrossRef]
10. Zhang, Y.; Li, N.; Zhang, L.; Lin, J.; Gao, X.; Chen, G. A review on the recent developments in vision-based apple-harvesting robots for recognizing fruit and picking pose. *Comput. Electron. Agric.* **2025**, *231*, 109968. [CrossRef]
11. Pathan, M.; Patel, N.; Yagnik, H.; Shah, M. Artificial cognition for applications in smart agriculture: A comprehensive review. *Artif. Intell. Agric.* **2020**, *4*, 81–95. [CrossRef]
12. Zhu, A.; Zhang, R.; Zhang, L.; Yi, T.; Wang, L.; Zhang, D.; Chen, L. YOLOv5s-CEDB: A robust and efficiency Camellia oleifera fruit detection algorithm in complex natural scenes. *Comput. Electron. Agric.* **2024**, *221*, 108984. [CrossRef]
13. Li, H.; Gu, Z.; He, D.; Wang, X.; Huang, J.; Mo, Y.; Li, P.; Huang, Z.; Wu, F. A lightweight improved YOLOv5s model and its deployment for detecting pitaya fruits in daytime and nighttime light-supplement environments. *Comput. Electron. Agric.* **2024**, *220*, 108914. [CrossRef]
14. Teng, H.; Sun, F.; Wu, H.; Lv, D.; Lv, Q.; Feng, F.; Yang, S.; Li, X. DS-YOLO: A Lightweight Strawberry Fruit Detection Algorithm. *Agronomy* **2025**, *15*, 2226. [CrossRef]
15. An, Q.; Wang, K.; Li, Z.; Song, C.; Tang, X.; Song, J. Real-Time Monitoring Method of Strawberry Fruit Growth State Based on YOLO Improved Model. *IEEE Access* **2022**, *10*, 124363–124372. [CrossRef]

16. Su, Y.; Xiong, J.; Tang, K.; Huang, Q.; Liao, K.; Wu, Z.; Zhang, M.; Peng, H. Identifying guava and its fruit stem in nighttime environment based on AP-UNet. *J. Food Compos. Anal.* **2025**, *140*, 107211. [CrossRef]
17. Ma, Z.; Dong, N.; Gu, J.; Cheng, H.; Meng, Z.; Du, X. STRAW-YOLO: A detection method for strawberry fruits targets and key points. *Comput. Electron. Agric.* **2025**, *230*, 109853. [CrossRef]
18. Li, P.; Wen, M.; Zeng, Z.; Tian, Y. Cherry Tomato Bunch and Picking Point Detection for Robotic Harvesting Using an RGB-D Sensor and a StarBL-YOLO Network. *Horticulturae* **2025**, *11*, 949. [CrossRef]
19. Zhao, G.; Dong, S.; Wen, J.; Ban, Y.; Zhang, X. Selective fruit harvesting prediction and 6D pose estimation based on YOLOv7 multi-parameter recognition. *Comput. Electron. Agric.* **2024**, *229*, 109815. [CrossRef]
20. Mei, Z.; Li, Y.; Zhu, R.; Wang, S. Intelligent Fruit Localization and Grasping Method Based on YOLO VX Model and 3D Vision. *Agriculture* **2025**, *15*, 1508. [CrossRef]
21. Rana, S.; Hensel, O.; Nasirahmadi, A. From vineyard to vision: Multi-domain analysis and mitigation of grape cluster detection failures in complex viticultural environments. *Results Eng.* **2026**, *29*, 108833. [CrossRef]
22. Liu, Z.; Rasika, D.; Abeyrathna, R.M.; Sampurno, R.M.; Nakaguchi, V.M.; Ahamed, T. A new occlusion-avoidance and dual-view fruit localization method with a 6-DoF manipulator for orchard harvesting. *Comput. Electron. Agric.* **2025**, *237*, 110634. [CrossRef]
23. Yang, L.; Noguchi, T.; Hoshino, Y. Development of a pumpkin fruits pick-and-place robot using an RGB-D camera and a YOLO based object detection AI model. *Comput. Electron. Agric.* **2024**, *227*, 109625. [CrossRef]
24. Xiao, S.; Zhao, Q.; Chen, Y.; Li, T. A dual-backbone lightweight detection and depth position picking system for multiple occlusions Camellia oleifera fruit. *Comput. Electron. Agric.* **2025**, *233*, 110157. [CrossRef]
25. Ma, B.L.; Hua, Z.X.; Wen, Y.C.; Deng, H.X.; Zhao, Y.J.; Pu, L.R.; Song, H.B. Using an improved lightweight YOLOv8 model for real-time detection of multi-stage apple fruit in complex orchard environments. *Artif. Intell. Agric.* **2024**, *11*, 70–82. [CrossRef]
26. Du, X.; Cheng, H.; Ma, Z.; Lu, W.; Wang, M.; Meng, Z.; Jiang, C.; Hong, F. DSW-YOLO: A detection method for ground-planted strawberry fruits under different occlusion levels. *Comput. Electron. Agric.* **2023**, *214*, 108304. [CrossRef]
27. Liu, Z.; Zhuo, L.; Dong, C.; Li, J. YOLO-TBD: Tea Bud Detection with Triple-Branch Attention Mechanism and Self-Correction Group Convolution. *Ind. Crops Prod.* **2025**, *226*, 120607. [CrossRef]
28. Murat, A.A.; Kiran, M.S. A comprehensive review on YOLO versions for object detection. *Eng. Sci. Technol. Int. J.* **2025**, *70*, 102161. [CrossRef]
29. GHoward, A.; Zhu, M.; Chen, B.; Kalenichenko, D.; Wang, W.; Weyand, T.; Andreetto, M.; Adam, H. MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications. *arXiv* **2017**, arXiv:1704.04861. [CrossRef]
30. Sandler, M.; Howard, A.; Zhu, M.; Zhmoginov, A.; Chen, L.; IEEE. MobileNetV2: Inverted Residuals and Linear Bottlenecks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–23 June 2018.
31. Howard, A.G.; Sandler, M.; Chen, B.; Wang, W.; Chen, L.-C.; Tan, M.; Chu, G.; Vasudevan, V.; Zhu, Y.; Pang, R.; et al. Searching for MobileNetV3. In *Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Korea, 27 October–2 November 2019*; IEEE: Piscataway, NJ, USA, 2019; pp. 1314–1324.
32. Enebuse, I.; Foo, M.; Ibrahim, B.S.K.K.; Ahmed, H.; Supmak, F.; Eyobu, O.S. A Comparative Review of Hand-Eye Calibration Techniques for Vision Guided Robots. *IEEE Access* **2021**, *9*, 113143–113155. [CrossRef]
33. Zhang, X.; Zhou, X.; Lin, M.; Sun, R.; IEEE. ShuffleNet: An Extremely Efficient Convolutional Neural Network for Mobile Devices. In Proceedings of the 31st IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–23 June 2018.
34. Ma, N.; Zhang, X.; Zheng, H.-T.; Sun, J. ShuffleNet V2: Practical Guidelines for Efficient CNN Architecture Design. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018.

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.