applied sciences

MDPI

Article

A Dual-Backbone Collaborative Feature Enhancement Network for Low-Contrast Small-Object Detection in Parallel Fruit-Sorting Robots

Jiawei Shao and Guoqin Gao *

School of Electrical & Information Engineering, Jiangsu University, Zhenjiang 212013, China; shaojiawei001@126.com
* Correspondence: gqgao@ujs.edu.cn

Abstract

Detecting grape stems in clustered fruits is challenging because of their small size, low contrast with surrounding fruits, and frequent occlusion. To address these challenges, this study proposes ELCD-YOLO, a dual-backbone collaborative feature-enhancement network for grape-stem detection in the visual perception stage of parallel fruit-sorting robots. First, YOLOv12s and EfficientNetV2-N are employed as heterogeneous parallel backbones to extract complementary semantic and local texture features from the same input image. Cross-backbone feature fusion is combined with an adapted lightweight cross-layer reconstruction (LR) module to reorganize the fused P3, P4, and P5 features into higher-resolution P2, P3, and P4 detection representations without increasing the number of detection heads. DySample is employed within the cross-scale reconstruction pathways for content-adaptive spatial alignment, while CAFM is serially introduced after A2C2 for complementary contextual–local feature refinement. These components are coordinated within a unified architecture to address the coupled challenges of small target size, low contrast, occlusion, and complex background interference. Experiments were conducted on a grape-stem detection dataset constructed primarily from images acquired using the robotic platform, with data augmentation restricted to the training set and identified acquisition-group overlaps removed from the evaluation subsets. Across three independent training runs, ELCD-YOLO achieved an mAP@0.5 of 68.00 ± 0.25%, Precision of 92.51 ± 0.25%, and Recall of 59.50 ± 0.35%, representing improvements of 2.89, 3.10, and 4.18 percentage points over YOLOv12s, respectively. The model requires 21.5 GFLOPs, occupies 23.6 MB, and achieves 81 FPS for detector-pipeline inference on an NVIDIA RTX 4070. These results demonstrate improved grape-stem detection performance while maintaining real-time inference capability, supporting further investigation of ELCD-YOLO as a visual perception component for parallel fruit-sorting robots.

Keywords: parallel fruit-sorting robot; object detection; low contrast; small objects; fruit-stem detection; dual-backbone network

Check for updates

Academic Editor: Yutaka Ishibashi

Received: 23 August 2026
Revised: 10 October 2026
Accepted: 24 October 2026
Published: 29 October 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

With the development of smart agriculture and agricultural robotics, automated fruit sorting has become an important component of intelligent agricultural product processing. In robotic sorting systems, the visual perception module is responsible for obtaining the locations and categories of fruits and graspable targets, and its detection accuracy and real-time performance directly affect subsequent grasp localization and sorting efficiency [1–4].

Compared with conventional manual sorting, machine-vision-based non-contact detection offers advantages such as strong adaptability, a high degree of automation, and relatively low cost, and has been widely applied to fruit detection, localization, robotic harvesting, and sorting tasks [5–10]. For clustered fruits such as grapes, the dense distribution of fruits results in severe occlusion and structural overlap, making direct fruit localization insufficient for stable robotic grasping in many cases. To reduce mechanical damage to fruits during grasping, fruit stems are commonly selected as grasping targets. Parallel robots provide high dynamic responsiveness and motion-control accuracy and are therefore suitable for high-speed fruit-sorting tasks. However, their practical performance strongly depends on the ability of the vision system to detect and localize fruit stems rapidly and accurately [11–14].

In recent years, deep-learning-based object detection methods have been widely adopted in agricultural robotic vision. Conventional two-stage detectors, such as R-CNN, Fast R-CNN, and Faster R-CNN, generally provide high detection accuracy, but their relatively high computational complexity limits their applicability to real-time robotic systems [15–17]. In contrast, one-stage detectors represented by SSD and the YOLO family provide a favorable balance between detection accuracy and computational efficiency and have therefore been extensively applied to industrial inspection, intelligent manufacturing, and agricultural robotic vision tasks [18–22]. With the continuous development of the YOLO family, its detection capability in complex environments has been substantially improved. Nevertheless, fruit-stem detection for parallel fruit-sorting robots remains challenging.

First, fruit stems typically occupy only a small region of an image and can therefore be regarded as typical small objects. As network depth increases, repeated downsampling enriches high-level semantic representations but simultaneously causes the gradual loss of shallow spatial details, boundary information, and texture features, thereby weakening small-object representation and localization. Second, grape fruits and stems often exhibit similar colors and textures, resulting in low contrast between the stems and surrounding fruits. Under such low-contrast conditions, weak stem boundaries and local texture cues can easily be overwhelmed by complex background information, leading to missed detections and localization errors. In addition, occlusion caused by the dense distribution of clustered fruits, illumination variations, and complex backgrounds further increases feature confusion between targets and their surroundings. Therefore, this task is not merely a small-object detection problem but rather a coupled visual perception problem involving small objects, low contrast, and occlusion interference.

Existing approaches generally improve detection performance through multi-scale feature fusion, attention mechanisms, lightweight network architectures, and feature enhancement strategies. However, under low-contrast small-object conditions, a single feature extraction pathway still has difficulty simultaneously preserving global semantic information and local texture details. Excessive reliance on deep semantic features may result in the loss of weak boundaries and fine-grained texture information, whereas excessive emphasis on local features may make the detector more susceptible to interference from complex backgrounds and occluded regions. Therefore, achieving complementary enhancement in global semantic and local texture information while maintaining low computational complexity is a key challenge for improving low-contrast small-object detection in parallel fruit-sorting robots.

To address these challenges, this study proposes ELCD-YOLO, a dual-backbone collaborative feature-enhancement network for low-contrast small-object detection. Built upon the YOLOv12s framework [23], ELCD-YOLO employs a heterogeneous dual-backbone architecture combining YOLOv12s and EfficientNetV2-N [24] to extract complementary semantic and local texture representations from the same input image. Corresponding

multi-scale features from the two branches are integrated through cross-backbone feature fusion to strengthen the representation of low-contrast small targets. A lightweight cross-layer reconstruction module (LR), adapted from the cross-layer reconstruction concept of LSOD-YOLO [25], is further incorporated into the dual-backbone framework to reorganize the fused P3, P4, and P5 features into higher-resolution P2, P3, and P4 detection representations without increasing the number of detection heads. DySample [26] is employed within the cross-scale reconstruction pathways to provide content-adaptive spatial alignment before feature fusion. In addition, the existing CAFM module [27] is serially placed after the original A2C2 block to provide complementary convolution–attention feature refinement while preserving the internal structure of A2C2. In the corresponding cross-scale pathways, the resulting processing sequence follows A2C2 → CAFM → DySample → Concat. Through these coordinated architectural adaptations, ELCD-YOLO aims to improve the representation and detection of low-contrast small objects while controlling computational complexity and maintaining real-time detector inference capability for visual perception applications in parallel fruit-sorting robots.

The main contributions of this study are summarized as follows:

1. A task-oriented heterogeneous dual-backbone collaborative feature-enhancement framework, ELCD-YOLO, is developed for low-contrast small-object detection. Rather than relying on a single hierarchical feature-extraction pathway, YOLOv12s and EfficientNetV2-N are employed as parallel branches to obtain complementary semantic and local texture representations from the same input image. Their corresponding multi-scale features are integrated through cross-backbone fusion, providing a richer representation of weak grape-stem targets under low-contrast, occluded, and complex-background conditions.
2. A high-resolution cross-layer reconstruction strategy is adapted to the heterogeneous dual-backbone fusion framework. Inspired by the cross-layer reconstruction concept of LSOD-YOLO [25], the lightweight LR-based neck operates on the fused P3, P4, and P5 representations and reorganizes them into P2, P3, and P4 detection features. Unlike approaches that simply add an additional high-resolution detection head, this configuration shifts the existing three-level detection representation toward higher spatial resolutions without increasing the number of detection heads, thereby improving the utilization of fine-grained spatial information while controlling computational complexity.
3. A coordinated feature-reconstruction and refinement pathway is established by integrating dynamic spatial alignment with serial contextual–local feature enhancement. DySample [26] is incorporated into the required cross-scale reconstruction pathways to provide content-adaptive spatial alignment before feature fusion, while CAFM [27] is serially placed after the original A2C2 block to refine local spatial and texture information without modifying the internal structure of A2C2. In the corresponding pathways, the processing sequence follows A2C2 → CAFM → DySample → Concat. The contribution therefore lies not in independently proposing these existing mechanisms but in coordinating them with heterogeneous feature extraction, cross-backbone fusion, and high-resolution reconstruction to address the coupled challenges of small target size, low contrast, occlusion, and background interference.
4. A unified evaluation framework is established on a grape-stem detection dataset constructed primarily from images acquired using the parallel fruit-sorting robot platform. The original images are partitioned before augmentation, augmentation is restricted to the training subset, and identified acquisition-group overlaps are removed from the evaluation subsets. Comparative and sequential ablation experiments are conducted under a common final evaluation protocol, with the principal configurations evaluated

across three independent training runs. The experimental validation is intentionally restricted to grape-stem detection and image-space bounding-box localization, providing a reproducible basis for assessing ELCD-YOLO as a visual perception component for parallel fruit-sorting applications.

2. Related Work and Research Motivation

2.1. Small-Object Detection

Small-object detection remains a challenging task in computer vision, mainly because small targets occupy only a limited number of pixels in an image, and their fine-grained spatial information can be progressively lost through repeated downsampling. Recent studies have mainly improved small-object representation and detection performance through multi-scale feature fusion, attention mechanisms, and lightweight network architectures.

Chen et al. [13] improved citrus detection performance in orchard environments by introducing additional detection layers and incorporating residual and dense connection structures. Zhao et al. [28] proposed a YOLOv5s-based detection model for tomato seedlings and introduced a contextual information module to enhance the representation of target texture features. Sun et al. [29] replaced the conventional backbone network with MobileNetV3 and incorporated coordinate attention to improve grape-cluster detection performance.

Although these studies have achieved promising results in small-object feature enhancement and model lightweighting, repeated downsampling during deep feature extraction may still cause the loss of fine-grained spatial information, particularly for small targets with weak texture cues. This can lead to missed detections and false positives. Therefore, enhancing fine-grained small-object representation while maintaining real-time detection efficiency remains an important research challenge.

2.2. Low-Contrast Object Detection

Low-contrast object detection represents another challenging problem in computer vision because the high visual similarity between targets and their surrounding backgrounds often results in ambiguous object boundaries and insufficient discriminative features. Existing studies have mainly addressed this problem through attention mechanisms, feature enhancement strategies, and network architecture optimization to improve the feature distinction between targets and backgrounds.

Hao et al. [30] improved walnut detection performance by replacing the YOLOv3 backbone with an attention-enhanced MobileNetV3 architecture. Shi et al. [31] incorporated the CBAM attention mechanism into YOLOv5 to enhance contextual feature perception and improve the detection of green citrus. Zhang et al. [32] proposed an improved YOLOv9-based detection framework that combines an efficient multi-scale attention mechanism with an optimized loss function to enhance detection performance in low-contrast traffic scenarios.

Although these approaches can improve the representation of low-contrast targets to some extent, existing studies mainly focus on attention enhancement, feature enhancement, or loss-function optimization. When targets exhibit both low contrast and small-object characteristics, a detector must not only distinguish weak feature differences between targets and backgrounds but also preserve the limited local spatial details of the targets. Therefore, simultaneously maintaining local detail information and global semantic representation remains an important challenge for object detection in complex low-contrast environments.

2.3. Robust Object Detection

Maintaining stable detection performance under complex conditions, including occlusion, background interference, and illumination variations, is another important research direction in object detection. To address feature interference in complex environments,

existing studies have mainly improved model robustness through attention mechanisms, background suppression, and feature enhancement strategies.

Lu et al. [33] proposed MASF-YOLO for small-object detection in UAV imagery by introducing an improved efficient multi-scale attention module to suppress background noise and improve detection performance in complex scenes. Ji et al. [34] developed YOLO-TLA, which employs a global attention mechanism to generate weighted feature maps, thereby enhancing target responses and suppressing irrelevant background information. Liu et al. [35] proposed DiffYOLO for PCB defect detection by incorporating denoised features generated by diffusion models to improve the model's resistance to complex noise.

Although these methods effectively improve robustness in complex environments, existing feature-fusion approaches still largely rely on channel-wise feature concatenation. While such operations can effectively aggregate multi-scale feature information, their ability to explicitly model spatial dependencies between features remains limited. When targets exhibit low contrast, occlusion, and texture similarity, conventional feature concatenation may not sufficiently establish spatial relationships between target regions and contextual information, making the targets more susceptible to interference from complex backgrounds.

2.4. Research Gap and Motivation

In summary, existing studies on small-object detection, low-contrast object detection, and robust object detection have achieved substantial progress through multi-scale feature fusion, attention mechanisms, lightweight architectures, and feature-enhancement strategies. However, several limitations remain when these methods are applied to low-contrast grape-stem detection in parallel fruit-sorting robots. First, conventional single-backbone detectors rely on a single hierarchical feature-extraction pathway, which may limit their ability to simultaneously preserve fine-grained local texture information and high-level semantic context. This limitation becomes particularly relevant when the targets are small and exhibit weak visual differences from the surrounding fruits. Second, repeated downsampling can progressively weaken boundary and spatial detail information, while subsequent fixed upsampling operations provide limited adaptability for recovering informative small-object features during cross-scale reconstruction. Third, under severe occlusion and complex fruit-cluster backgrounds, contextual modeling alone may be insufficient to preserve weak local cues, whereas conventional feature-fusion operations do not explicitly address the complementary refinement of contextual and local spatial representations. Therefore, an effective detection framework for this task should jointly exploit heterogeneous feature representations, enhance high-resolution multi-scale information, provide adaptive cross-scale spatial alignment, and refine weak local features while maintaining manageable computational complexity and real-time inference capability.

Several existing approaches address individual aspects of these challenges, but their architectural emphasis differs from that of the present study. LSOD-YOLO [25] introduces a lightweight cross-layer reconstruction strategy for small-object detection and provides the conceptual basis for the reconstruction mechanism adapted in this work. In contrast, the present study applies cross-layer reconstruction to features that have first been obtained through heterogeneous dual-backbone extraction and cross-backbone fusion. Specifically, the fused P3, P4, and P5 representations are reorganized into P2, P3, and P4 detection features without increasing the number of detection heads, thereby shifting the three-level detection representation toward higher spatial resolutions. CSF-Net [36], which is also based on a dual-backbone small-object detection framework, provides another relevant comparison. The architectural distinction of ELCD-YOLO lies in its specific integration of heterogeneous dual-backbone feature extraction and fusion, high-resolution cross-layer reconstruction, dynamic spatial alignment, and serial contextual–local feature refinement

within a unified detection pipeline. This task-oriented configuration differentiates the proposed framework from the related small-object detection approaches considered in this study while building upon previously established architectural mechanisms.

Based on these considerations, this study proposes ELCD-YOLO, a dual-backbone collaborative feature-enhancement network for low-contrast small-object detection. A heterogeneous dual-backbone architecture combining YOLOv12s and EfficientNetV2-N is employed to extract complementary semantic and local texture representations from the same input image, followed by cross-backbone fusion at the P3, P4, and P5 levels. An adapted lightweight cross-layer reconstruction module (LR) then reorganizes the fused multi-scale features into higher-resolution P2, P3, and P4 detection representations without increasing the number of detection heads. DySample is incorporated into the required cross-scale reconstruction pathways to provide content-adaptive spatial alignment before feature fusion. In addition, CAFM is serially placed after the original A2C2 block to provide complementary convolution–attention feature refinement while preserving the internal structure of A2C2. Accordingly, the primary contribution of ELCD-YOLO lies in the task-oriented adaptation and integration of established architectural mechanisms, rather than in proposing fundamentally new component-level methods. Specifically, heterogeneous feature extraction, cross-backbone fusion, high-resolution cross-layer reconstruction, dynamic spatial alignment, and serial contextual–local feature refinement are incorporated into an application-oriented detection framework designed to address the combined challenges of small target size, low contrast, occlusion, and complex background interference in grape-stem detection. This configuration is intended to improve detection performance while maintaining manageable computational complexity, with its effectiveness evaluated under the task-specific experimental conditions considered in this study.

3. Materials and Methods

3.1. Overall Architecture of ELCD-YOLO

To address the challenges of low-contrast small-object detection in vision-guided parallel fruit-sorting robots, this study proposes ELCD-YOLO (Efficient Low-Contrast Detection YOLO), a dual-backbone collaborative feature-enhancement network developed on the basis of YOLOv12s. In grape-stem detection, the targets are typically small, exhibit weak texture and boundary cues, and have high visual similarity to the surrounding fruits. These characteristics, together with frequent occlusion and complex backgrounds, can weaken target representation and increase the difficulty of accurate detection and localization.

Conventional single-backbone detectors rely on a single hierarchical feature-extraction pathway and may therefore have difficulty simultaneously preserving fine-grained local information and high-level semantic context. For low-contrast small targets, repeated downsampling can progressively weaken shallow spatial details and boundary information, while conventional fixed upsampling operations may provide limited flexibility for recovering spatially informative features during cross-scale fusion. In addition, contextual modeling alone may not be sufficient to preserve weak local cues under severe occlusion and background interference. These considerations motivate the coordinated use of complementary feature extraction, cross-layer reconstruction, adaptive spatial alignment, and local feature refinement.

Accordingly, ELCD-YOLO modifies the YOLOv12s framework from four complementary aspects. First, a heterogeneous dual-backbone architecture composed of the YOLOv12s and EfficientNetV2-N backbones is employed to extract complementary semantic and local texture representations from the same input image. Corresponding multi-scale features from the two branches are fused through DFF-based cross-backbone fusion at the P3, P4, and P5 levels.

Second, a Lightweight Cross-layer Reconstruction Module (LR), adapted from the cross-layer reconstruction concept of LSOD-YOLO [25], is incorporated as the neck of the proposed network. The fused P3, P4, and P5 features are reorganized through cross-layer reconstruction to generate the final P2, P3, and P4 detection features, thereby shifting the three-level detection representation toward higher spatial resolutions without increasing the number of detection heads.

Third, DySample [26] is employed in the required cross-scale reconstruction pathways to provide content-adaptive spatial alignment before feature concatenation. This replaces the corresponding fixed nearest-neighbor upsampling operations in the complete ELCD-YOLO configuration and is intended to improve the preservation of spatial details during multi-scale feature reconstruction.

Finally, CAFM [27] is serially placed after the original A2C2 block to provide complementary convolution–attention feature refinement while preserving the internal structure of A2C2. Thus, the basic feature-enhancement sequence is A2C2 → CAFM, while the corresponding cross-scale reconstruction pathways follow A2C2 → CAFM → DySample → Concat. This configuration allows contextual information modeled by A2C2 and local spatial representations refined by CAFM to be successively utilized before cross-scale feature fusion.

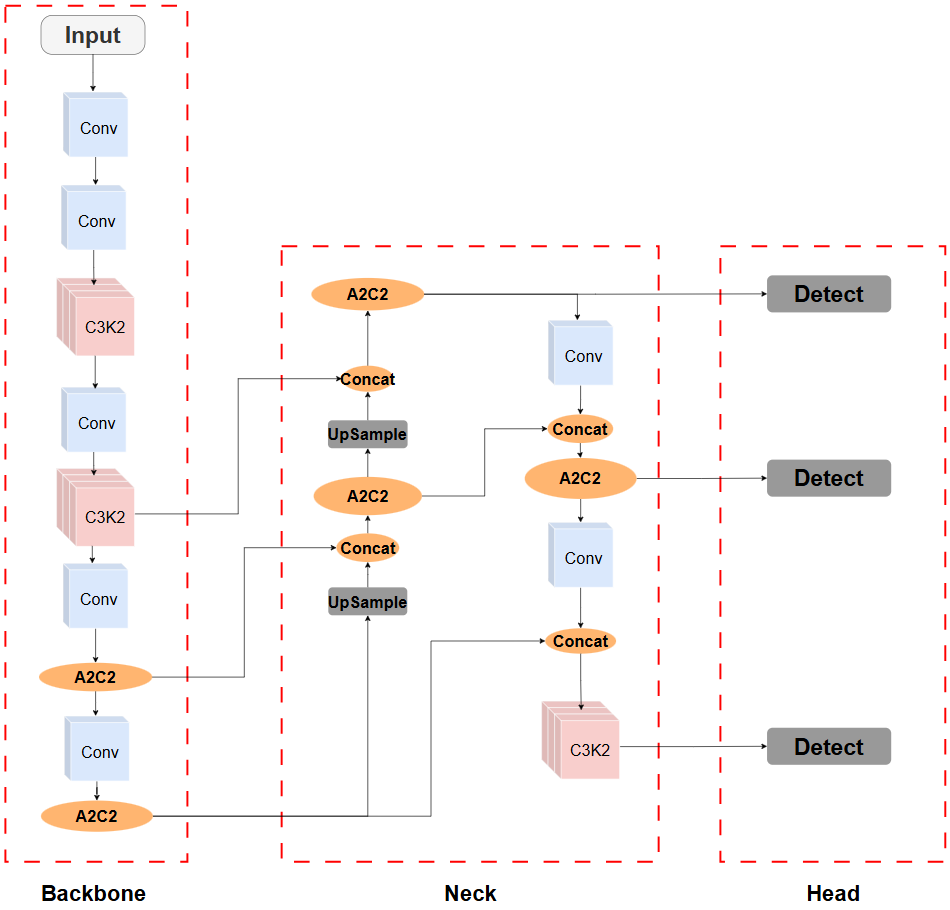
The overall architectures of the YOLOv12s baseline and ELCD-YOLO are illustrated in Figure 1. For an input resolution of 640 × 640, the DFF-fused P3, P4, and P5 features are further reconstructed by the LR-based neck into three detection features at P2, P3, and P4, with dimensions of 160 × 160 × 128, 80 × 80 × 128, and 40 × 40 × 256, respectively. These three feature maps are directly supplied to the corresponding Detect heads. Through the coordinated integration of heterogeneous feature extraction, cross-backbone fusion, cross-layer reconstruction, dynamic spatial alignment, and serial feature refinement, ELCD-YOLO is designed to improve the representation of low-contrast grape-stem targets while maintaining a favorable balance between detection performance and computational complexity.

For further architectural details, the stage-wise configuration of the principal cross-backbone fusion, cross-layer reconstruction, feature-enhancement, and detection pathways is provided in Supplementary Table S3.

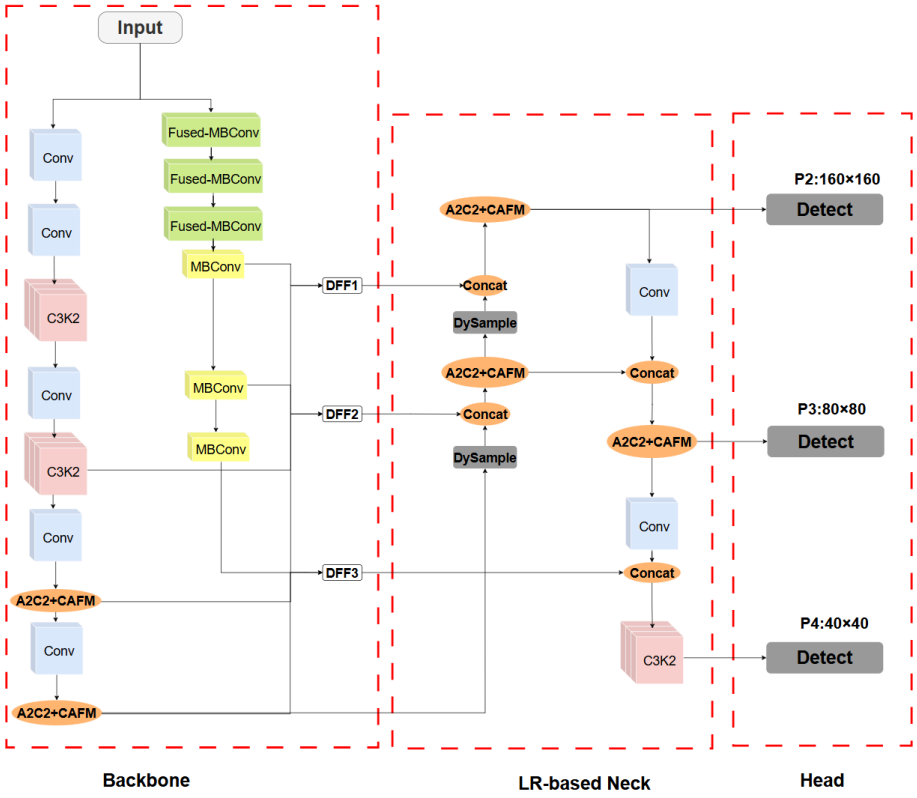
3.2. Dual-Backbone Feature Extraction Structure

Feature representation capability plays a critical role in low-contrast small-object detection. In grape-stem detection scenarios, targets are typically characterized by small sizes, weak texture information, and high color similarity to surrounding fruits. Conventional single-backbone networks are required to simultaneously capture global semantic information and local texture details. However, the different receptive-field requirements and feature characteristics of these two types of information may limit the ability of a single feature extraction pathway to adequately preserve both, particularly when the target boundaries are weak and the available spatial information is limited.

The YOLOv12s backbone employs the Area Attention mechanism to model long-range spatial dependencies and capture contextual information [23]. However, repeated downsampling may weaken shallow texture and boundary information, making the detection of small targets with limited visual features more challenging. In grape-stem detection scenarios, where the targets exhibit high visual similarity to surrounding fruits, preserving fine-grained local information alongside global semantic representations is important for improving target discrimination and reducing missed detections.



(**a**) Structure of YOLOv12s Model



(**b**) Structure of ELCD-YOLO Model

Figure 1. Comparison of Network Architectures Between ELCD-YOLO Model and YOLOv12s Model.

To address these challenges, a heterogeneous dual-backbone feature extraction architecture with same-source parallel input is developed, as illustrated in Figure 1b. The same input image is simultaneously fed into two parallel feature extraction branches, namely the EfficientNetV2-N branch and the YOLOv12s branch. These heterogeneous architectures are employed to extract complementary feature representations at different levels, enabling the integration of local texture details and global contextual information for low-contrast small-object detection.

Branch A serves as the complementary detail extraction pathway and adopts EfficientNetV2-N as the auxiliary backbone [24]. EfficientNetV2 incorporates Fused-MBConv and MBConv structures for efficient feature extraction. In the proposed architecture, this branch is employed to provide complementary local texture, edge, and spatial-detail representations, supporting the preservation of fine-grained information for low-contrast grape-stem targets.

Branch B serves as the primary semantic extraction pathway and retains the original YOLOv12s backbone architecture. Its Area Attention mechanism enables the modeling of contextual relationships among spatial regions, providing semantic information for identifying potential target regions. By retaining the original YOLOv12s backbone, the proposed framework maintains the hierarchical feature extraction structure of the baseline detector while introducing complementary representations from the auxiliary branch.

The feature maps extracted by the two backbones are subsequently integrated through cross-backbone feature fusion at three corresponding feature levels, namely P3, P4, and P5. The corresponding feature pairs, denoted as C3/C3′, C4/C4′, and C5/C5′, are processed by DFF-based cross-stage fusion modules to integrate complementary representations from the two branches. The resulting fused feature maps contain 128, 128, and 256 channels, respectively, and are further processed by the lightweight cross-layer reconstruction (LR) module for subsequent multi-scale feature enhancement and detection.

Through the coordinated extraction and integration of heterogeneous features, the proposed dual-backbone architecture is designed to improve the representation of low-contrast small objects while controlling the additional computational complexity introduced by the auxiliary backbone. Its effectiveness in grape-stem detection is evaluated through the comparative and ablation experiments presented in Section 5.

3.3. Lightweight Cross-Layer Output Module

In YOLO-based object detectors, small-object detection performance is often constrained by the progressive loss of spatial detail during repeated downsampling. Although deeper feature maps provide stronger semantic representations, shallow boundary, texture, and positional information may gradually weaken as the network depth increases. This issue is particularly relevant to grape-stem detection, where the targets are small, low-contrast, and frequently occluded. Under these conditions, insufficient high-resolution spatial information may lead to missed detections and inaccurate localization.

A common strategy for improving small-object detection is to introduce an additional high-resolution P2 detection head. Although this approach can preserve more fine-grained spatial information, it also increases the computational cost associated with high-resolution feature processing. In addition, high-resolution feature maps may contain considerable background information, which can introduce additional interference in low-contrast scenes. Therefore, rather than increasing the number of detection heads, this study reorganizes the multi-scale feature-output pathway to enhance the utilization of high-resolution information while maintaining three detection levels.

To this end, a Lightweight Cross-layer Reconstruction Module (LR) is incorporated into ELCD-YOLO, as illustrated in Figure 2c. The design of LR is adapted from the cross-layer

reconstruction concept of the LCOR module in LSOD-YOLO [25]. Rather than presenting the cross-layer reconstruction strategy itself as a new principle, the present study adapts this concept to the heterogeneous dual-backbone architecture of ELCD-YOLO. Specifically, the LR module operates on the multi-scale features obtained after cross-backbone fusion and reconstructs them into detection features that are more suitable for low-contrast small-object detection.

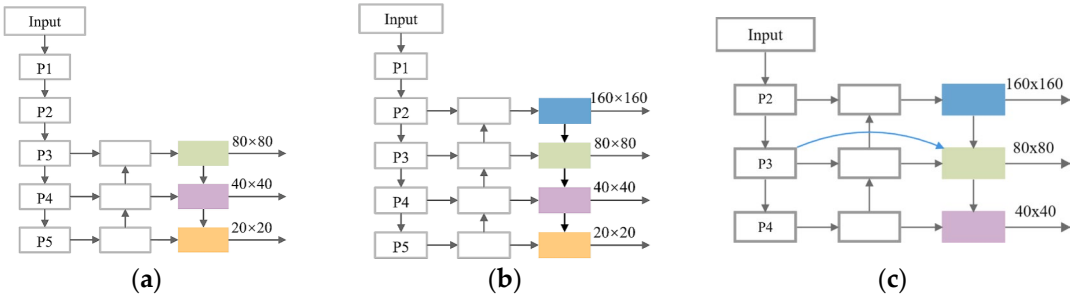


Figure 2. Detection layer structures of different networks. (**a**) The network detection layer structure of YOLOv12s. (**b**) The network detection layer structure of YOLOv12s with added P2 layer detection head. (**c**) The structure of lightweight cross-layer reconstruction module (LR).

The corresponding features extracted from the YOLOv12s and EfficientNetV2-N branches are first fused by the DFF modules at the P3, P4, and P5 levels. These fused multi-scale features are then introduced into the LR-based neck for cross-layer reconstruction. Through lateral connections and cross-scale information propagation, high-resolution spatial details are combined with deeper semantic representations. In the complete ELCD-YOLO configuration, DySample [26] is used in the required upsampling pathways to perform content-adaptive spatial alignment before feature concatenation. The aligned features are subsequently fused through channel-wise concatenation to strengthen the interaction between spatial details and semantic information across different scales.

After cross-layer reconstruction, the LR-based neck produces three final detection features corresponding to P2, P3, and P4, with dimensions of 160 × 160 × 128, 80 × 80 × 128, and 40 × 40 × 256, respectively, for an input resolution of 640 × 640. These three reconstructed feature maps are directly supplied to the corresponding Detect heads. Therefore, the proposed configuration does not increase the number of detection heads; instead, it shifts the three-level detection representation toward higher spatial resolutions to improve the representation of small grape-stem targets.

Although the overall cross-layer reconstruction strategy is conceptually related to LCOR [25], its implementation in ELCD-YOLO is adapted to the proposed dual-backbone feature-fusion framework. In particular, the LR module receives DFF-fused heterogeneous backbone features, operates jointly with the proposed P3/P4/P5 cross-backbone fusion pathway, and produces the P2/P3/P4 detection features used by the final detector. In the complete model, the reconstruction process is further combined with DySample-based dynamic alignment. These adaptations enable the cross-layer reconstruction concept to be integrated into the proposed ELCD-YOLO architecture for low-contrast grape-stem detection.

The contribution of the LR module is evaluated through the ablation experiments presented in Section 5. The results are used to assess whether the adapted cross-layer reconstruction pathway improves small-object feature representation while maintaining a favorable balance between detection performance and computational complexity.

3.4. CAFM-Enhanced Feature Processing

Attention mechanisms are widely employed in object detection networks to enhance feature representation by emphasizing informative spatial regions and modeling contextual relationships. The A2C2 module employed in YOLOv12s [23] incorporates area-based attention to capture long-range dependencies among spatial regions and provides contextual feature modeling within the backbone and neck. However, for low-contrast small-object detection, contextual modeling alone may be insufficient to preserve weak boundary cues, local texture information, and fine-grained spatial structures. This limitation is particularly relevant to grape-stem detection, where the targets are small, frequently occluded, and visually similar to the surrounding fruits.

To provide complementary local feature enhancement, the Convolution and Attention Fusion Module (CAFM) proposed in previous research [27] is introduced into the ELCD-YOLO framework. CAFM combines attention-based feature modeling with convolutional feature extraction to integrate contextual information and local spatial representations. Its original structure is illustrated in Figure 3. In this study, CAFM is adopted as an existing feature-enhancement mechanism rather than being presented as a newly developed attention module.

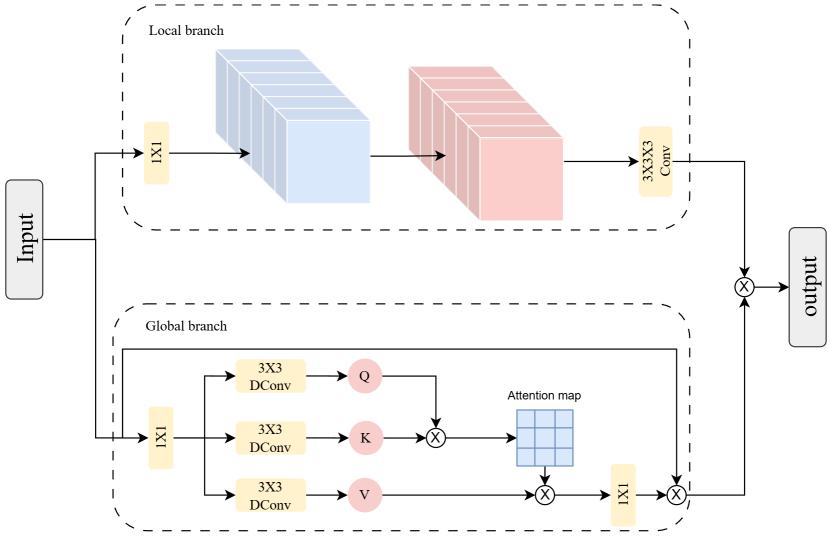


Figure 3. The structure of CAFM.

Importantly, CAFM is not embedded within or used to modify the internal structure of the original A2C2 block. Instead, the term A2C2 + CAFM used in this study denotes a composite feature-processing configuration in which CAFM is serially placed after the original A2C2 block. Thus, the basic processing sequence is A2C2 → CAFM, where A2C2 first performs area-attention-based contextual modeling and the subsequent CAFM module further refines the resulting features by combining convolutional local representations with attention-based contextual information. This serial configuration preserves the original A2C2 structure while introducing complementary feature enhancement without redefining its internal architecture.

Within ELCD-YOLO, the serial A2C2+CAFM configuration is applied to the multi-scale feature-processing pathways of the proposed network. After contextual features are generated by A2C2, CAFM further enhances local texture, boundary, and spatial information that may otherwise be weakened under low-contrast and occluded conditions. For the cross-scale reconstruction pathways in the LR-based neck, the CAFM-refined features are subsequently processed by DySample for spatial alignment and then concatenated with the

corresponding lateral features. Therefore, the local processing sequence in these pathways can be summarized as A2C2 → CAFM → DySample → Concat.

This configuration allows contextual information extracted by A2C2 and complementary local representations refined by CAFM to be successively utilized before cross-scale feature fusion.

The purpose of introducing CAFM is therefore not to propose a new attention principle, but to adapt an existing convolution–attention fusion mechanism [27] to the feature-processing pipeline of the proposed dual-backbone detection framework. By serially combining A2C2 and CAFM, the network is designed to improve the representation of weak grape-stem features under low-contrast, occluded, and complex-background conditions while maintaining a clear separation between the original A2C2 structure and the adopted CAFM module.

The contribution of this configuration is evaluated through the ablation experiments presented in Section 5. In particular, the comparison between EL-YOLO and ELC-YOLO is used to assess the effect of introducing CAFM after the A2C2-based feature-processing stage in terms of detection accuracy and computational complexity.

3.5. DySample Dynamic Upsampling Module

In YOLO-based object detection networks, feature pyramid architectures commonly rely on upsampling operations to facilitate information fusion across different feature scales. YOLOv12s employs nearest-neighbor interpolation for feature-map upsampling because of its low computational cost and implementation simplicity. However, nearest-neighbor interpolation is inherently a fixed sampling strategy, where sampling locations are predetermined and cannot adapt to image content.

For low-contrast small-object detection tasks, target regions typically contain limited texture and boundary information. During the upsampling process, fixed sampling locations may fail to accurately preserve informative target features, resulting in blurred boundaries, distorted texture structures, and discontinuous spatial representations. This issue becomes particularly severe in grape stem detection scenarios, where stems exhibit high visual similarity to surrounding fruits and naturally possess weak boundary cues. Consequently, conventional interpolation strategies may further degrade fine-grained feature reconstruction and reduce discriminative capability in subsequent detection layers.

To address these challenges, DySample, a previously proposed dynamic upsampling method [26], is adopted in the ELCD-YOLO framework, as illustrated in Figure 4. Unlike conventional interpolation methods that rely on fixed sampling locations, DySample formulates upsampling as a dynamic point-sampling process in which sampling positions are adjusted according to the input feature content. In the proposed architecture, DySample is incorporated into the cross-layer feature reconstruction pathways of the LR module to replace conventional upsampling operations in the complete ELCD-YOLO configuration. This integration enables content-adaptive spatial alignment of lower-resolution features before their fusion with features at corresponding spatial resolutions.

Specifically, DySample first predicts content-aware spatial offsets from the input feature map and then dynamically adjusts sampling locations based on these learned offsets. As a result, the generated sampling points can focus more accurately on informative target regions. By allowing sampling positions to vary according to feature distributions, DySample preserves boundary textures and spatial structures more effectively, thereby improving feature reconstruction quality.

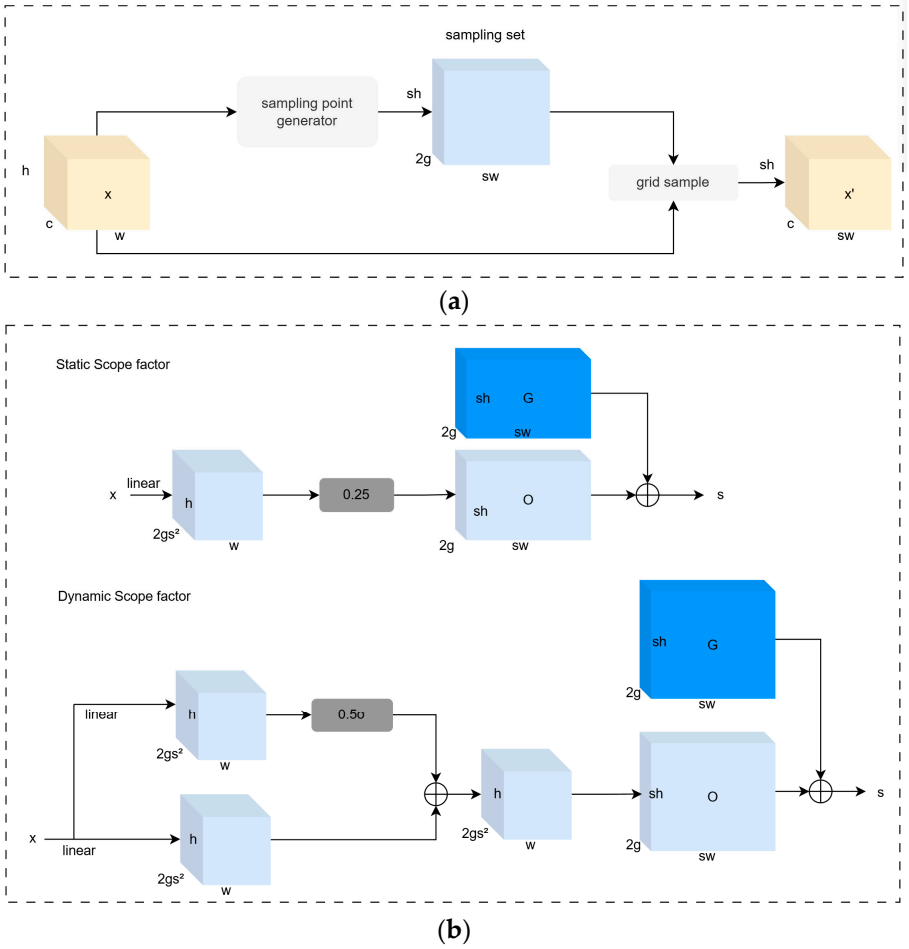


Figure 4. The network structure of DySample. (**a**) Sampling-based dynamic upsampling. (**b**) Sampling point generator in DySample.

Compared with dynamic-convolution-based or deformable-convolution-based upsampling approaches, DySample does not require additional kernel-generation networks or complex parameter-learning processes. Instead, it achieves adaptive feature reconstruction through an efficient point-based dynamic sampling mechanism, maintaining low computational overhead while enhancing detail preservation.

By incorporating DySample into the cross-scale reconstruction pathways, ELCD-YOLO is designed to provide content-adaptive spatial alignment and improve the utilization of fine-grained feature information. The sequential ablation results show improved mAP@0.5, Precision, and Recall after introducing DySample into the preceding ELC-YOLO configuration. However, boundary preservation and spatial-alignment quality were not independently evaluated; therefore, the observed detection improvements should not be interpreted as direct quantitative evidence of these specific mechanisms.

4. Experimental Platform and Dataset

4.1. Parallel Fruit Sorting Robot Platform

The proposed ELCD-YOLO model is developed for grape-stem detection in the visual perception stage of parallel fruit-sorting applications. A self-developed parallel fruit-sorting robot platform [37] provides the application context for this study. The platform mainly consists of an industrial computer, a control unit, a binocular vision system, a parallel robotic manipulator, and a grasping end-effector, as illustrated in Figure 5.

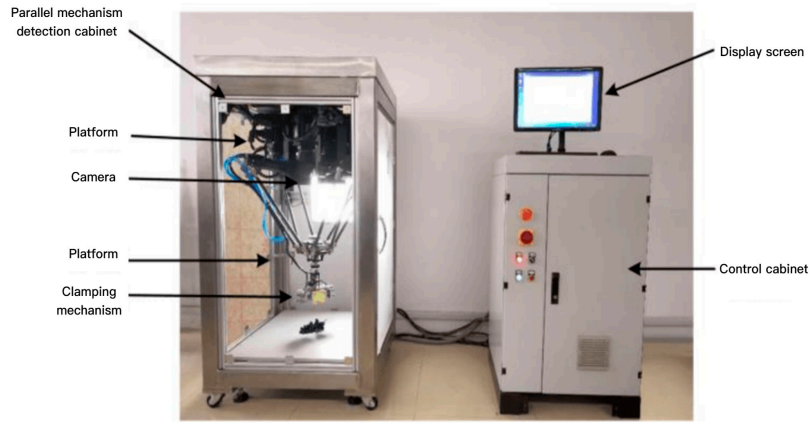


Figure 5. Clustered fruit-sorting parallel robot system.

Considering the requirements for visual perception and target localization in fruit-sorting applications, the platform adopts an Eye-to-Hand vision configuration [38], in which the binocular camera is fixed outside the robot workspace to reduce viewpoint disturbances caused by robot motion. The vision system provides image data for detecting grape stems, which can serve as potential grasping targets for subsequent robotic operations. In the present study, ELCD-YOLO is employed to identify grape stems in camera images and output their corresponding bounding boxes in image coordinates. These detection results provide visual information that may be used in subsequent target localization and robotic manipulation processes.

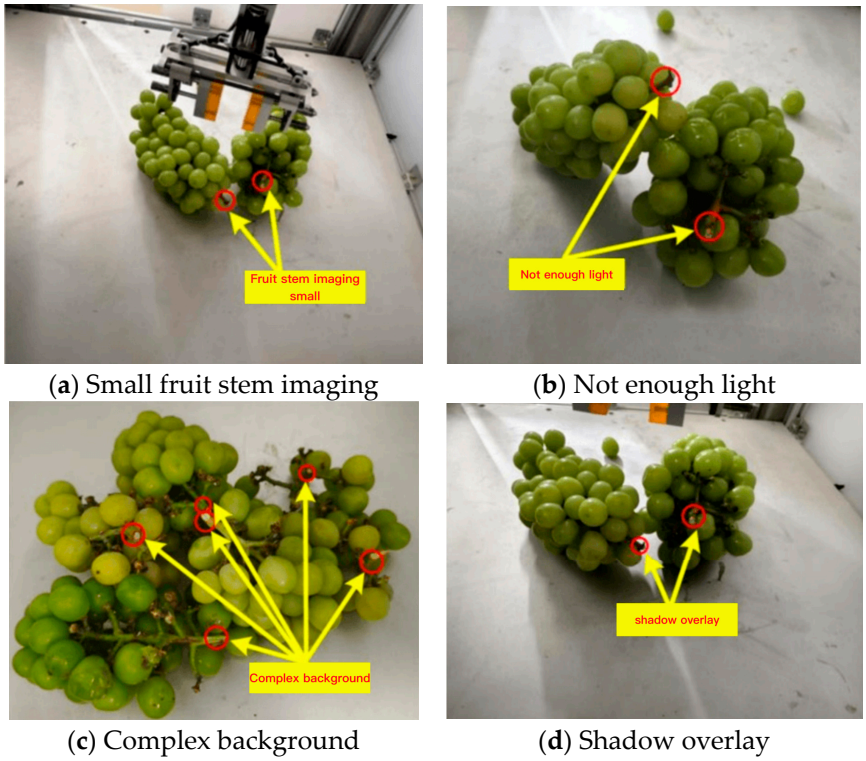
In a complete robotic sorting system, the detected image-space target positions would need to be converted into robot-coordinate positions through appropriate camera calibration, three-dimensional localization, and coordinate transformation procedures before being transmitted to the motion controller for grasping and sorting operations. However, the present study focuses on the visual detection component, and these downstream robotic operations are not quantitatively evaluated. Specifically, no experiments measuring robot-coordinate localization accuracy, grasping success rate, sorting success rate, or complete robotic cycle time are reported. Therefore, the experimental results presented in this study characterize the detection accuracy and inference performance of ELCD-YOLO rather than the operational performance of the complete robotic sorting system. Closed-loop robotic manipulation and quantitative validation of the complete sorting process remain subjects for future research.

4.2. Dataset Construction

Existing publicly available object-detection datasets do not adequately represent the visual characteristics of low-contrast grape stems in clustered fruit-sorting scenarios. In particular, grape stems typically occupy only a small region of the image and exhibit weak boundaries, high visual similarity to surrounding fruits, and frequent occlusion. These characteristics make it difficult to distinguish stem targets from complex backgrounds. To evaluate the proposed ELCD-YOLO under such conditions, a dedicated grape-stem detection dataset was assembled.

The dataset contained 1096 original images in total. Most of the images were acquired using the binocular vision system of the parallel fruit-sorting robot platform, while a small number of publicly sourced grape images were already included within the original 1096-image pool and were assigned exclusively to the test subset. The publicly sourced images were never used for training or validation. Images acquired using the robotic platform had a resolution of 1920 × 1080 pixels. The acquisition conditions covered different grape varieties, illumination conditions, and stacking configurations, including challenging scenarios

involving low target–background contrast, insufficient illumination, dense grape clusters, shadow interference, and partial occlusion. Most platform images were independently captured, while a smaller proportion originated from continuous acquisition or extracted video frames. Representative examples under different imaging conditions are shown in Figure 6.



(**a**) Small fruit stem imaging (**b**) Not enough light

(**c**) Complex background (**d**) Shadow overlay

Figure 6. Multi-scene clustered fruit image data.

To reduce the risk of information leakage between the training and evaluation subsets, the dataset was partitioned at the original-image level before any data augmentation was performed. The 1096 original images were initially divided into 877 training images, 109 validation images, and 110 test images, corresponding approximately to an 8:1:1 split.

Because a subset of the platform images originated from continuous acquisition or extracted video frames, the validation and test subsets were further screened for potential source-related overlap with the training subset. Since the original files had previously been mixed and renamed, acquisition-group membership could not be determined solely from filenames. The screening therefore combined the available acquisition information with conservative visual inspection. Images were considered to belong to the same acquisition group when one or more of the following criteria were satisfied:

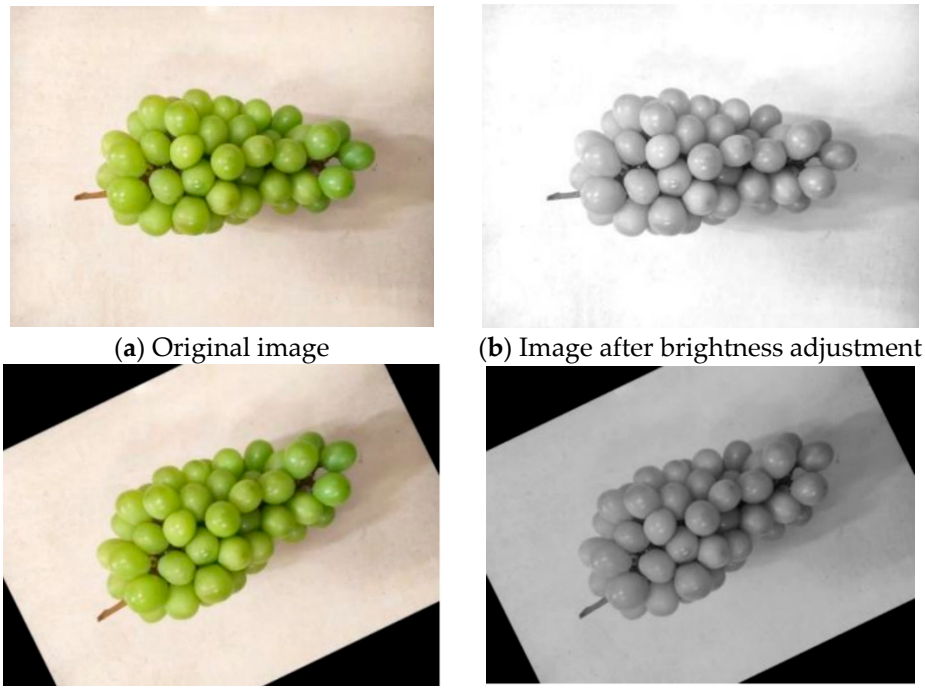
(1) The images originated from the same identifiable video or continuous acquisition sequence;
(2) The images depicted the same identifiable physical grape cluster, as determined from grape-cluster geometry, stem arrangement, surrounding background, and other recognizable visual characteristics, including cases captured from different viewpoints;
(3) The images corresponded to adjacent or near-consecutive video frames with substantially overlapping scene content.

Using these criteria, four validation images (0097, 0299, 0547, and 0782) and two test images (0237 and 0971) were identified as belonging to acquisition groups already represented in the training subset and were excluded from the corresponding evaluation

subsets. Images 0097 and 0782 depicted the same identifiable physical grape clusters as images in the training subset, whereas images 0237, 0299, 0547, and 0971 originated from the same video acquisition sequences as training images. After this screening, the validation and test subsets contained 105 and 108 images, respectively. No augmented version of a training image was included in either evaluation subset.

Additional information on the dataset composition, image-source categories, and the six excluded evaluation images is provided in Supplementary Table S1.

Data augmentation was subsequently applied only to the training subset to increase training-sample diversity. The augmentation operations included random rotation, horizontal flipping, scaling, illumination adjustment, and noise perturbation. Following augmentation, the 877 original training images were expanded to 4731 training images, whereas the validation and test subsets remained unaugmented. The validation subset was used for model selection, and the final test subset was used for independent performance evaluation. Examples of the augmentation operations are shown in Figure 7. Under the final evaluation protocol, the dataset therefore consisted of 4731 training images, 105 validation images, and 108 test images.



(**a**) Original image (**b**) Image after brightness adjustment

(**c**) Image after 26° rotation (**d**) Rotated and adjusted brightness image

Figure 7. Image data augmentation of clustered fruits.

All images were manually annotated using LabelImg (version 1.8.6). The detection task contained only one annotated object category, stem, with bounding boxes assigned to visible grape-stem targets. Although grape clusters were present in the images, they were treated as scene content rather than as a separate detection category. Therefore, all models were trained and evaluated as single-class grape-stem detectors. The final dataset contained 19,743 annotated stem instances in the augmented training subset, 486 instances in the validation subset, and 543 instances in the test subset. The image counts and annotated stem instances for the final dataset are summarized in Table 1.

Table 1. Dataset composition and annotation statistics.

Subset	Original Images After Initial Split	Final Images Used	Stem Instances	Augmentation
train	877	4731	19,743	yes
val	109	105	486	no
test	110	108	543	no

Note: The original 1096-image pool consisted predominantly of images acquired using the robotic experimental platform together with a small number of publicly sourced grape images included only in the test subset. Four validation images and two test images were excluded after acquisition-group overlap screening.

To further characterize the small-object properties of the grape-stem targets, a subset selected from the previously shuffled, unaugmented annotation pool, comprising 1808 stem bounding boxes, was analyzed. The median normalized bounding-box width was 0.0354, with an interquartile range (IQR) of 0.0229–0.0448, while the median normalized height was 0.0515, with an IQR of 0.0404–0.0735. The median normalized bounding-box area was 0.1685% of the image area, with an IQR of 0.1180–0.2390%. The observed minimum–maximum ranges were 0.0094–0.0969 for normalized width, 0.0147–0.1599 for normalized height, and 0.0373–0.7843% for normalized area.

These statistics are intended to provide a descriptive indication of the small-object characteristics of the sampled unaugmented stem annotations and are not presented as exact distributional statistics for the complete dataset. Together with the low target–background contrast, occlusion, illumination variation, and complex fruit-cluster backgrounds represented in the dataset, these target-size characteristics provide a challenging evaluation setting for low-contrast grape-stem detection.

4.3. Experimental Settings

All experiments were conducted under a Windows 10 operating system. The hardware platform consisted of an Intel Core i7 processor (Intel Corporation, Santa Clara, CA, USA), 32 GB RAM, and an NVIDIA GeForce RTX 4070 GPU with 12 GB memory (NVIDIA Corporation, Santa Clara, CA, USA). The proposed model was implemented using the PyTorch deep learning framework (version 2.1.0) with the CUDA 11.8 runtime.

For the principal ELCD-YOLO training and ablation experiments, a common training configuration was adopted to ensure comparability. The input image size was set to 640 × 640 pixels. Stochastic Gradient Descent (SGD) was used as the optimizer, with an initial learning rate of 0.01, a momentum coefficient of 0.937, and a weight decay coefficient of 0.0005. The training batch size was set to 4, and each model was trained for 300 epochs. In ELCD-YOLO, both the YOLOv12s and EfficientNetV2-N backbone branches were trained from scratch without pretrained weights, and no layers were frozen during training.

All detection models reported in the comparative experiments were evaluated using the same final 108-image test set after acquisition-group overlap screening. For previously trained comparison models whose training data and single-class configuration were unchanged, the existing trained checkpoints were re-evaluated on this final test set rather than being retrained. The specific retraining and re-evaluation status of the comparison models is described in Section 5.1.

To evaluate run-to-run variability, the YOLOv12s, E-YOLO, EL-YOLO, ELC-YOLO, and ELCD-YOLO configurations were each independently trained three times using random seeds of 0, 1200, and 2400. All three runs for a given configuration used the same dataset partition, model structure, and training hyperparameters. For each independent run, the checkpoint achieving the highest validation mAP@0.5 was selected as the best model and

subsequently evaluated on the final 108-image test set. The corresponding detection results are reported as the mean ± standard deviation across the three independent runs. The individual results obtained using the three random seeds, together with the corresponding mean and sample standard deviation, are provided in Supplementary Table S2.

For inference-speed evaluation, no separate timing dataset was used. The final 108-image test set was repeatedly traversed in a fixed cyclic order to form a sequence of 500 timed inputs. The same image sequence and ordering were used for all compared models. This 500-input sequence was measured 10 times, resulting in 5000 image-level timing observations. All speed measurements were performed at an input resolution of 640 × 640 pixels, with a batch size of 1 and FP32 numerical precision.

Before formal timing, three untimed warm-up passes were performed to reduce initialization-related fluctuations, and GPU synchronization was applied around the timed GPU operations. The measured latency included image preprocessing, network forward inference, and non-maximum suppression (NMS) post-processing. Under these conditions, ELCD-YOLO achieved an inference speed of 81 frames per second (FPS) on the NVIDIA GeForce RTX 4070 GPU.

The reported FPS therefore represents the throughput of the detector pipeline rather than that of the complete robotic fruit-sorting system. Image acquisition, data transmission, camera-to-robot coordinate transformation, three-dimensional target localization, motion planning, robot motion, and end-effector operation were excluded from the timing measurements.

4.4. Evaluation Metrics

To comprehensively evaluate the performance of the proposed ELCD-YOLO model for low-contrast small-object detection, Precision (P), Recall (R), and mean Average Precision at an intersection-over-union threshold of 0.50 (mAP@0.5) were adopted as the primary detection-accuracy metrics. In addition, model size, giga floating-point operations (GFLOPs), and frames per second (FPS) were used to characterize storage requirements, computational cost, and detector-pipeline inference efficiency, respectively.

Precision measures the proportion of correctly detected targets among all positive predictions, while Recall measures the proportion of ground-truth targets that are successfully detected. They are defined as follows:

$$Recall = \frac{TP}{TP + FN} \times 100\% \quad (1)$$

$$Precision = \frac{TP}{TP + FP} \times 100\% \quad (2)$$

where (TP) (True Positive) denotes a correctly detected target, (FP) (False Positive) denotes an incorrect positive prediction, and (FN) (False Negative) denotes a ground-truth target that is not successfully detected. A predicted bounding box is considered a true positive when it is correctly matched to a ground-truth bounding box with an intersection-over-union (IoU) value of at least 0.50.

For the reported Precision and Recall values, the confidence threshold was fixed at 0.45, and the non-maximum suppression (NMS) IoU threshold was set to 0.50 for all evaluated models. The IoU threshold used for matching predictions to ground-truth boxes and the IoU threshold used for NMS serve different purposes, although both were set to 0.50 in this study. The same evaluation settings were applied to all compared models to ensure consistency of the reported results.

Average Precision (AP) summarizes detection performance by integrating Precision over the corresponding Recall range. Unlike Precision and Recall evaluated at a fixed

confidence threshold, AP characterizes detector performance across different confidence levels. At an IoU threshold of 0.50, AP is expressed as follows:

$$AP@0.5 = \int_0^1 P(R)dR \tag{3}$$

where (P(r)) denotes Precision as a function of Recall (r) along the precision–recall curve.

Mean Average Precision (mAP) is obtained by averaging the AP values across all annotated detection categories. In this study, the reported metric is explicitly defined as mAP@0.5, rather than mAP@0.5:0.95, and is calculated as follows:

$$mAP@0.5 = \frac{\sum_{k=1}^{K} AP_k}{K}@0.5 \tag{4}$$

where (K) denotes the number of annotated detection categories and (AP_k@0.5) represents the Average Precision of category (k) at an IoU threshold of 0.50. Since the dataset contains only one annotated detection category, namely stem ($K = 1$), mAP@0.5 is numerically equivalent to AP@0.5 for the stem category. Grape clusters are treated as scene content and are not evaluated as a separate detection category.

In addition to detection accuracy, model size, GFLOPs, and FPS were used to characterize the computational characteristics of the evaluated models. Model size represents the storage space occupied by the saved model, while GFLOPs denotes the floating-point computational cost of a forward pass at an input resolution of 640 × 640 pixels. FPS represents the number of images processed per second under the specified inference conditions. As described in Section 4.3, FPS was measured on an NVIDIA GeForce RTX 4070 GPU with a batch size of 1 and FP32 precision. The measured detector-pipeline latency included image preprocessing, network forward inference, and NMS post-processing, while image acquisition, data transmission, camera-to-robot coordinate transformation, three-dimensional localization, motion planning, and robotic manipulation were excluded. Therefore, the reported FPS reflects the inference efficiency of the detection pipeline rather than the operational throughput of the complete robotic fruit-sorting system.

For the principal baseline and sequential ablation configurations—YOLOv12s, E-YOLO, EL-YOLO, ELC-YOLO, and ELCD-YOLO—Precision, Recall, and mAP@0.5 are reported as the mean ± sample standard deviation across three independent training runs using random seeds of 0, 1200, and 2400. The individual seeded-run results are provided in Supplementary Table S2 to characterize run-to-run variability and improve the reproducibility of the reported results.

5. Results

5.1. Overall Performance Comparison

To evaluate the detection performance and computational characteristics of ELCD-YOLO for low-contrast small-object detection, comparative experiments were conducted using several representative YOLO-based detectors, including YOLOv5s, YOLOv8s, YOLOv8l, YOLOv11s, and YOLOv12s. In addition, LSOD-YOLO [25], a lightweight small-object detector, and CSF-Net [36], a dual-backbone small-object detection network, were included to provide comparisons with architectures developed for small-object detection.

To ensure a consistent evaluation protocol, all models reported in Table 2 were evaluated on the same final 108-image test set after acquisition-group overlap screening. Because the training subset and the single-class stem configuration remained unchanged after the screening procedure, the previously trained checkpoints of YOLOv5s, YOLOv8s, YOLOv8l, and YOLOv11s were not retrained. Instead, these checkpoints were re-evaluated on the final 108-image test set, and the corresponding detection results were updated. LSOD-YOLO

and CSF-Net were also evaluated on the same final test set and are reported as single-run comparison results.

Table 2. Comparison of different object detection algorithms on the final 108-image test set.

Model	mAP@0.5 (%)	Precision (%)	Recall (%)	GFLOPs	Model Size (MB)	FPS
YOLOv5s	57.82	81.23	49.47	16.3	14.5	120
YOLOv8s	63.74	85.44	52.72	28.6	11.2	98
YOLOv8l	68.15	91.42	56.14	165.7	43.69	57
YOLOv11s	64.62	87.83	54.28	21.6	21.5	95
YOLOv12s	65.11 ± 0.30	89.41 ± 0.25	55.32 ± 0.25	21.5	19.8	86
LSOD-YOLO [25]	65.92	82.71	58.90	33.9	7.6	93
CSF-Net [36]	65.60	85.01	60.10	18.5	16.2	90
ELCD-YOLO	68.00 ± 0.25	92.51 ± 0.25	59.50 ± 0.35	21.5	23.6	81

Note: YOLOv12s and ELCD-YOLO are reported as the mean ± standard deviation over three independent training runs using random seeds of 0, 1200, and 2400. YOLOv5s, YOLOv8s, YOLOv8l, YOLOv11s, LSOD-YOLO, and CSF-Net are reported as single-run results. All models were evaluated on the same final 108-image test set. Numerical differences involving single-run models should therefore be interpreted cautiously because their run-to-run variability was not estimated.

For the principal baseline and proposed model, YOLOv12s and ELCD-YOLO were each independently trained three times using random seeds of 0, 1200, and 2400. Their mAP@0.5, Precision, and Recall values are therefore reported as the mean ± standard deviation across the three independent runs. The evaluation metrics are defined in Section 4.4. GFLOPs, model size, and FPS are additionally reported to characterize computational complexity, storage requirements, and detector-pipeline inference efficiency.

As shown in Table 2, the re-evaluated YOLOv5s checkpoint achieved an mAP@0.5 of 57.82%, a Precision of 81.23%, and a Recall of 49.47% on the final 108-image test set. The corresponding re-evaluated mAP@0.5 values for YOLOv8s, YOLOv8l, and YOLOv11s were 63.74%, 68.15%, and 64.62%, respectively. Their Precision values were 85.44%, 91.42%, and 87.83%, while their Recall values were 52.72%, 56.14%, and 54.28%, respectively. These models were not retrained because the training subset and class configuration remained unchanged; only their evaluation was repeated using the final common test set.

Among the YOLO-based comparison models, YOLOv8l yielded a single-run mAP@0.5 of 68.15%, together with 165.7 GFLOPs, a model size of 43.69 MB, and an inference speed of 57 FPS. ELCD-YOLO achieved a three-run mean mAP@0.5 of 68.00 ± 0.25%, with 21.5 GFLOPs, a model size of 23.6 MB, and an inference speed of 81 FPS. The reported mAP@0.5 of YOLOv8l was numerically 0.15 percentage points higher than the mean value of ELCD-YOLO. However, ELCD-YOLO required substantially fewer computational operations and less model storage and achieved a higher detector-pipeline inference speed under the present experimental conditions. Because YOLOv8l is represented by a single-run result, the small numerical difference in mAP@0.5 should be interpreted cautiously rather than as a statistically established performance difference.

Compared with the YOLOv12s baseline, ELCD-YOLO increased mAP@0.5 from 65.11 ± 0.30% to 68.00 ± 0.25%, Precision from 89.41 ± 0.25% to 92.51 ± 0.25%, and Recall from 55.32 ± 0.25% to 59.50 ± 0.35%. These changes correspond to improvements of 2.89, 3.10, and 4.18 percentage points, respectively. The reported computational complexity remained unchanged at 21.5 GFLOPs, whereas the model size increased from 19.8 MB to 23.6 MB and the inference speed decreased from 86 FPS to 81 FPS. These results indicate that the coordinated architectural modifications improve grape-stem detection performance

relative to the YOLOv12s baseline, accompanied by a moderate increase in model size and a small reduction in detector-pipeline inference speed.

To further assess ELCD-YOLO against methods specifically developed for small-object detection, comparisons were conducted with LSOD-YOLO [25] and CSF-Net [36]. LSOD-YOLO achieved single-run values of 65.92% mAP@0.5, 82.71% Precision, and 58.90% Recall, together with 33.9 GFLOPs, a model size of 7.6 MB, and an inference speed of 93 FPS. Relative to these single-run results, ELCD-YOLO showed numerical increases of 2.08 percentage points in mAP@0.5, 9.80 percentage points in Precision, and 0.60 percentage points in Recall while requiring fewer GFLOPs. LSOD-YOLO, however, had a substantially smaller model size and higher inference speed. These results indicate that the two architectures emphasize different trade-offs. More importantly, the distinction between the two approaches is architectural rather than simply numerical: LSOD-YOLO introduces cross-layer reconstruction within a lightweight small-object detection framework, whereas ELCD-YOLO adapts this reconstruction concept to heterogeneous dual-backbone fused features and reorganizes the fused P3, P4, and P5 representations into P2, P3, and P4 detection features without increasing the number of detection heads.

CSF-Net achieved single-run values of 65.60% mAP@0.5, 85.01% Precision, and 60.10% Recall, with 18.5 GFLOPs, a model size of 16.2 MB, and an inference speed of 90 FPS. Relative to CSF-Net, ELCD-YOLO showed numerical increases of 2.40 percentage points in mAP@0.5 and 7.50 percentage points in Precision, whereas its mean Recall was 0.60 percentage points lower. CSF-Net also required fewer computational operations, occupied less storage space, and achieved a higher inference speed. Although both methods employ dual-backbone architectures for small-object detection, ELCD-YOLO does not rely on dual-backbone feature fusion alone. Its heterogeneous fused features are further processed through high-resolution cross-layer reconstruction, DySample-based content-adaptive spatial alignment, and serial A2C2+CAFM contextual–local feature refinement. Therefore, the comparison with CSF-Net highlights a different architectural strategy for exploiting dual-backbone information rather than establishing uniform superiority across all performance metrics. Because LSOD-YOLO and CSF-Net are represented by single-run results, the numerical differences relative to the repeated-run ELCD-YOLO results should be interpreted as descriptive comparisons rather than statistically established differences between methods.

The increase in Recall from 55.32 $\pm$ 0.25% for YOLOv12s to 59.50 $\pm$ 0.35% for ELCD-YOLO indicates that the proposed architecture detected a larger proportion of annotated grape stems under the present evaluation protocol. This improvement is relevant to the low-contrast small-object detection task, in which weak boundary information, small target size, occlusion, and visual similarity between grape stems and surrounding fruits can contribute to missed detections. Nevertheless, CSF-Net yielded a single-run Recall of 60.10%; therefore, the present results do not indicate that ELCD-YOLO achieves the highest Recall among all evaluated architectures. Rather, the experimental results support the effectiveness of the coordinated architectural design relative to the YOLOv12s baseline while revealing different accuracy–efficiency trade-offs compared with other specialized small-object detectors.

Overall, the comparative experiments show that ELCD-YOLO improves mAP@0.5, Precision, and Recall relative to the YOLOv12s baseline while retaining the same reported computational complexity and real-time detector-pipeline inference capability. Comparisons with YOLOv8l, LSOD-YOLO, and CSF-Net further reveal different trade-offs among detection accuracy, computational complexity, model size, and inference speed. The contribution of ELCD-YOLO should therefore be interpreted not as the independent introduction of dual-backbone extraction, cross-layer reconstruction, CAFM, or DySample, but as their task-oriented coordination within a unified low-contrast small-object detection framework. Specifically, heterogeneous feature extraction and cross-backbone fusion are

followed by P3/P4/P5-to-P2/P3/P4 high-resolution reconstruction, serial A2C2+CAFM feature refinement, and DySample-based dynamic spatial alignment. The sequential ablation results presented in Section 5.2 further evaluate the incremental effects of these coordinated architectural modifications. The reported results characterize image-level grape-stem detection and bounding-box localization and do not directly establish improvements in robot-coordinate localization accuracy, grasping success rate, sorting efficiency, or complete robotic system performance.

5.2. Ablation Study

5.2.1. Core Ablation Study

To quantitatively evaluate the effects of the major architectural modifications in ELCD-YOLO, a series of sequential ablation experiments were conducted using YOLOv12s as the baseline detector. The evaluated configurations progressively incorporated the heterogeneous dual-backbone architecture, the adapted lightweight cross-layer reconstruction module (LR), the serial A2C2+CAFM feature-enhancement configuration, and DySample-based dynamic upsampling. In the model names, E denotes the introduction of EfficientNetV2-N as the auxiliary backbone, L denotes the LR module, C denotes the serial placement of CAFM after the original A2C2 block, and D denotes the introduction of DySample in the corresponding cross-scale reconstruction pathways.

To assess run-to-run variability and reproducibility, YOLOv12s, E-YOLO, EL-YOLO, ELC-YOLO, and ELCD-YOLO were each independently trained three times using random seeds of 0, 1200, and 2400. All configurations used the same dataset partition, training settings, and final 108-image test set. For each independent run, the checkpoint achieving the highest validation mAP@0.5 was selected and subsequently evaluated on the final test set. The detection accuracy metrics are reported as the mean ± standard deviation across the three independent runs. The individual results for the three random seeds, together with the corresponding mean and sample standard deviation, are provided in Supplementary Table S2. The ablation results are summarized in Table 3.

Table 3. Results of the core sequential ablation experiments.

Model	Dual Backbone	LR	A2C2+CAFM	DySample	mAP@0.5 (%)	P (%)	R (%)	Size (MB)	GFLOPs
YOLOv12s	✗	✗	✗	✗	65.11 ± 0.30	89.41 ± 0.25	55.32 ± 0.25	19.8	21.5
E-YOLO	✓	✗	✗	✗	65.92 ± 0.20	91.61 ± 0.25	56.40 ± 0.35	27.6	23.2
EL-YOLO	✓	✓	✗	✗	66.30 ± 0.25	91.40 ± 0.30	57.50 ± 0.25	23.9	21.3
ELC-YOLO	✓	✓	✓	✗	66.62 ± 0.20	91.90 ± 0.30	58.61 ± 0.25	23.8	21.5
ELCD-YOLO	✓	✓	✓	✓	68.00 ± 0.25	92.51 ± 0.25	59.50 ± 0.35	23.6	21.5

Note: mAP@0.5, Precision, and Recall are reported as the mean ± standard deviation over three independent training runs using random seeds of 0, 1200, and 2400. The individual seeded-run results are provided in Supplementary Table S2. The A2C2+CAFM column denotes a serial configuration in which CAFM is placed after the original A2C2 block rather than being integrated into the internal structure of A2C2.

As shown in Table 3, introducing the heterogeneous dual-backbone architecture increased the mean mAP@0.5 from 65.11 ± 0.30% for YOLOv12s to 65.92 ± 0.20% for E-YOLO, corresponding to an increase of 0.81 percentage points. Mean Precision increased from 89.41% to 91.61%, while mean Recall increased from 55.32% to 56.40%, corresponding to increases of 2.20 and 1.08 percentage points, respectively. These results indicate that the addition of EfficientNetV2-N as a heterogeneous auxiliary feature-extraction branch is associated with improved detection performance relative to the YOLOv12s baseline under the present experimental conditions. However, the dual-backbone configuration increased

the model size from 19.8 MB to 27.6 MB and the computational complexity from 21.5 to 23.2 GFLOPs, indicating additional storage and computational requirements.

After introducing the adapted LR module, EL-YOLO achieved an mAP@0.5 of 66.30 ± 0.25%, representing a further increase of 0.38 percentage points over E-YOLO. Mean Recall increased from 56.40% to 57.50%, corresponding to an increase of 1.10 percentage points, whereas mean Precision decreased slightly from 91.61% to 91.40%, a reduction of 0.21 percentage points. Meanwhile, the model size decreased from 27.6 MB to 23.9 MB, and the reported computational complexity decreased from 23.2 to 21.3 GFLOPs. These results indicate that, within the heterogeneous dual-backbone configuration, the adapted cross-layer reconstruction pathway improves mAP@0.5 and Recall while reducing part of the additional model complexity introduced by the auxiliary backbone. The slight reduction in Precision also indicates that the LR-based modification does not independently improve every detection metric.

When CAFM was serially placed after the original A2C2 block, the resulting ELC-YOLO configuration achieved an mAP@0.5 of 66.62 ± 0.20%, a Precision of 91.90 ± 0.30%, and a Recall of 58.61 ± 0.25%. Relative to EL-YOLO, these mean values increased by 0.32, 0.50, and 1.11 percentage points, respectively. The model size decreased slightly from 23.9 MB to 23.8 MB, while the reported computational complexity increased from 21.3 to 21.5 GFLOPs. These incremental changes indicate that serial CAFM-based feature refinement after A2C2 is associated with additional detection gains within the current configuration. In this arrangement, the original A2C2 block remains structurally unchanged, while CAFM provides subsequent convolution–attention feature refinement before the corresponding cross-scale reconstruction operations.

Finally, introducing DySample into the ELC-YOLO configuration produced the complete ELCD-YOLO model, which achieved an mAP@0.5 of 68.00 ± 0.25%, a Precision of 92.51 ± 0.25%, and a Recall of 59.50 ± 0.35%. Relative to ELC-YOLO, the corresponding mean values increased by 1.38, 0.61, and 0.89 percentage points, respectively. The reported computational complexity remained unchanged at 21.5 GFLOPs, while the model size decreased slightly from 23.8 MB to 23.6 MB. These results show that replacing the corresponding fixed upsampling operations with DySample in the complete configuration is associated with improved detection metrics without increasing the reported computational complexity. The observed gains are consistent with the intended role of content-adaptive spatial alignment during cross-scale feature reconstruction, although the present ablation experiments do not independently quantify boundary reconstruction or spatial-alignment quality.

Overall, the sequential ablation experiments show a progressive improvement in detection performance from the YOLOv12s baseline to the complete ELCD-YOLO configuration. Mean mAP@0.5 increased from 65.11% to 68.00%, Precision from 89.41% to 92.51%, and Recall from 55.32% to 59.50%, corresponding to overall increases of 2.89, 3.10, and 4.18 percentage points, respectively. The complete model increased the model size from 19.8 MB to 23.6 MB, whereas the reported computational complexity remained at 21.5 GFLOPs, the same value as the YOLOv12s baseline. Taken together, the results indicate that the sequential integration of heterogeneous dual-backbone feature extraction, the adapted LR reconstruction pathway, serial A2C2+CAFM feature refinement, and DySample-based dynamic spatial alignment is associated with incremental detection improvements under the present experimental conditions. However, because the ablation experiments employ sequential configurations rather than a complete factorial evaluation, they do not independently establish interaction effects or synergistic gains among the individual components. The ablation results are further illustrated in Figure 8.

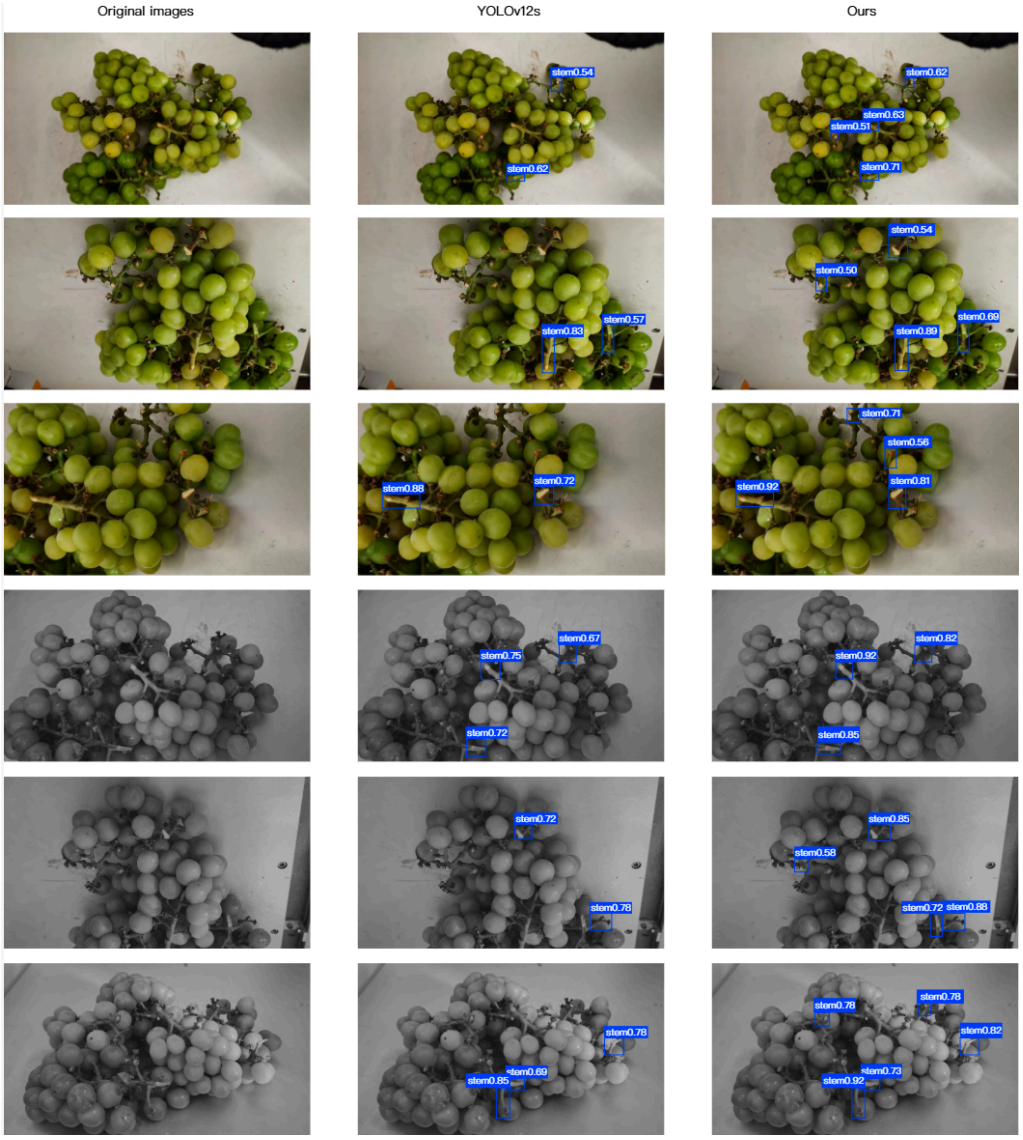


Figure 8. Qualitative comparison of grape-stem detection results between YOLOv12s and ELCD-YOLO under different imaging conditions.

5.2.2. Comparison of Different Auxiliary Backbone Architectures

To further investigate the influence of auxiliary-backbone selection on low-contrast small-object detection, comparative experiments were conducted using four lightweight backbone architectures: MobileNetV4, GhostNet, ShuffleNetV1, and EfficientNetV2-N. In all dual-backbone configurations, the original YOLOv12s backbone was retained as the primary feature-extraction branch, while only the auxiliary backbone was replaced. The adapted LR module, the serial A2C2+CAFM feature-enhancement configuration, DySample-based dynamic upsampling, and the detection-head configuration were kept unchanged so that the comparison focused on the effect of auxiliary-backbone selection.

All configurations were evaluated on the same final 108-image test set. The previously trained YOLO-M, YOLO-G, and YOLO-S checkpoints were re-evaluated on this final test set without retraining because the training subset and single-class configuration remained unchanged. These three configurations are therefore reported as single-run results. The EfficientNetV2-N-based complete ELCD-YOLO configuration is reported as the mean ± standard deviation over three independent training runs using random seeds of 0, 1200, and

2400. The YOLOv12s baseline is included as a reference and is likewise reported over three independent runs. The results are summarized in Table 4.

Table 4. Comparison of different auxiliary-backbone configurations.

Model	Auxiliary Backbone	mAP@0.5 (%)	P (%)	R (%)	Model Size (MB)	GFLOPs
YOLOv12s	None	65.11 ± 0.30	89.41 ± 0.25	55.32 ± 0.25	19.8	21.5
YOLO-M	MobileNetV4	65.21	89.72	55.13	20.1	20.9
YOLO-G	GhostNet	66.02	90.70	56.70	22.9	21.3
YOLO-S	ShuffleNetV1	65.91	91.70	56.40	23.7	21.4
ELCD-YOLO	EfficientNetV2-N	68.00 ± 0.25	92.51 ± 0.25	59.50 ± 0.35	23.6	21.5

Note: YOLOv12s and ELCD-YOLO are reported as the mean ± standard deviation over three independent training runs using random seeds of 0, 1200, and 2400. YOLO-M, YOLO-G, and YOLO-S are reported as single-run results obtained by re-evaluating their previously trained checkpoints on the same final 108-image test set. Numerical differences involving the single-run configurations should therefore be interpreted descriptively because their run-to-run variability was not estimated.

As shown in Table 4, the MobileNetV4-based configuration, YOLO-M, achieved an mAP@0.5 of 65.21%, a Precision of 89.72%, and a Recall of 55.13%. Relative to the three-run mean of the YOLOv12s baseline, these values correspond to numerical changes of +0.10, +0.31, and −0.19 percentage points, respectively. The model size was 20.1 MB, and the computational complexity was 20.9 GFLOPs. Under the present experimental conditions, the MobileNetV4-based configuration therefore maintained relatively low computational complexity but produced only small numerical changes in the detection metrics relative to the baseline.

When GhostNet was employed as the auxiliary backbone, YOLO-G achieved an mAP@0.5 of 66.02%, a Precision of 90.70%, and a Recall of 56.70%. Compared with the single-run YOLO-M result, mAP@0.5, Precision, and Recall increased numerically by 0.81, 0.98, and 1.57 percentage points, respectively. The model size increased from 20.1 MB to 22.9 MB, while the computational complexity increased from 20.9 to 21.3 GFLOPs. These results show that the GhostNet-based configuration produced higher reported detection metrics than the MobileNetV4-based configuration, accompanied by modest increases in model size and computational cost.

Using ShuffleNetV1 as the auxiliary backbone, YOLO-S achieved an mAP@0.5 of 65.91%, a Precision of 91.70%, and a Recall of 56.40%, with a model size of 23.7 MB and a computational complexity of 21.4 GFLOPs. Relative to YOLO-G, Precision increased by 1.00 percentage point, whereas mAP@0.5 and Recall decreased by 0.11 and 0.30 percentage points, respectively. The ShuffleNetV1-based configuration therefore exhibited a different balance among the three detection metrics, with higher Precision but slightly lower mAP@0.5 and Recall than the GhostNet-based configuration.

When EfficientNetV2-N was employed as the auxiliary backbone in the complete ELCD-YOLO configuration, the model achieved an mAP@0.5 of 68.00 ± 0.25%, a Precision of 92.51 ± 0.25%, and a Recall of 59.50 ± 0.35% across three independent training runs. Relative to the single-run MobileNetV4-, GhostNet-, and ShuffleNetV1-based configurations, the mean mAP@0.5 was numerically higher by 2.79, 1.98, and 2.09 percentage points, respectively. The corresponding Precision differences were 2.79, 1.81, and 0.81 percentage points, while the Recall differences were 4.37, 2.80, and 3.10 percentage points, respectively. The EfficientNetV2-N-based configuration had a model size of 23.6 MB and a computational complexity of 21.5 GFLOPs.

These results show that the EfficientNetV2-N-based configuration produced the highest reported values of mAP@0.5, Precision, and Recall among the evaluated auxiliary-backbone configurations under the present experimental protocol. However, because MobileNetV4-, GhostNet-, and ShuffleNetV1-based configurations are represented by single-run results whereas ELCD-YOLO is reported over three independent runs, the numerical differences should be interpreted as descriptive comparisons rather than statistically established differences among the auxiliary backbones.

EfficientNetV2 employs Fused-MBConv and MBConv structures for feature extraction. Within the proposed heterogeneous dual-backbone framework, EfficientNetV2-N is used to provide feature representations complementary to those extracted by the YOLOv12s branch. The observed detection results are consistent with the architectural motivation for introducing a heterogeneous auxiliary backbone. However, the present comparison does not independently measure local-texture preservation, boundary representation, or the contribution of individual internal structures within each backbone. Therefore, the observed differences should be interpreted as the performance of the evaluated auxiliary-backbone configurations within the complete detection framework rather than as direct evidence that a particular internal feature-extraction mechanism is responsible for the observed gains.

Overall, the auxiliary-backbone comparison supports the use of EfficientNetV2-N as the auxiliary feature-extraction branch in the present ELCD-YOLO configuration. Under the evaluated grape-stem detection conditions, the EfficientNetV2-N-based configuration achieved a three-run mean mAP@0.5 of 68.00 ± 0.25%, a Precision of 92.51 ± 0.25%, and a Recall of 59.50 ± 0.35%, with 21.5 GFLOPs and a model size of 23.6 MB. The observed performance differences remain specific to the present dataset, network configurations, and evaluation protocol.

Figure 9 further presents qualitative detection results obtained using the different auxiliary-backbone configurations. In the specific examples illustrated in Figure 9, the EfficientNetV2-N-based configuration produces more stem detections than the other compared configurations, including fewer missed detections in some low-contrast and occluded cases. These qualitative observations are limited to the illustrated examples and should not be interpreted as evidence of generally superior localization capability across different imaging conditions.

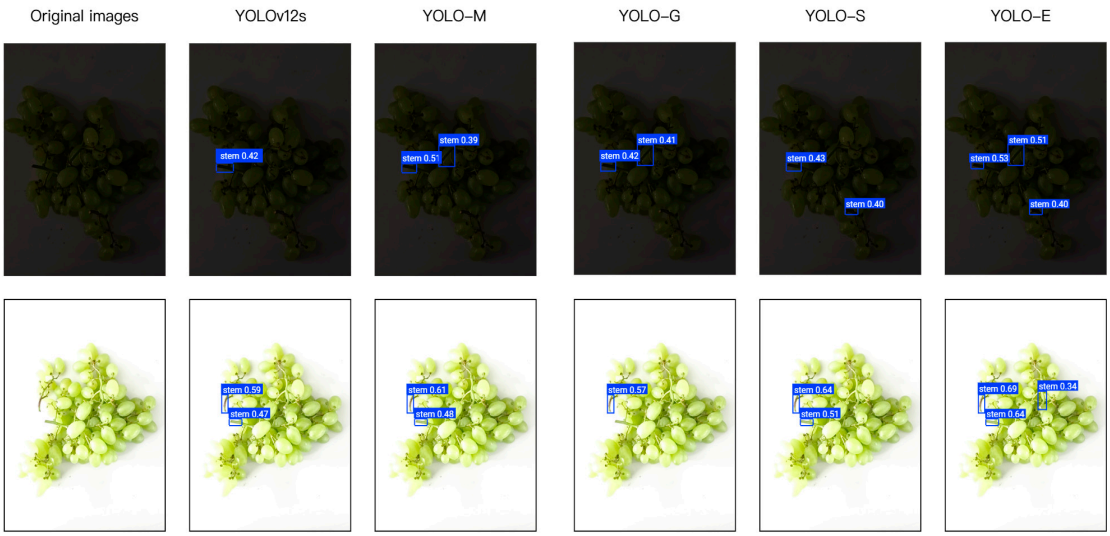


Figure 9. Detection results of models with different backbone combinations.

6. Conclusions

This study proposed ELCD-YOLO, a dual-backbone collaborative feature-enhancement network for low-contrast small-object detection, with a particular focus on grape-stem detection in the visual perception stage of parallel fruit-sorting robots. The proposed framework was developed to address the combined challenges of small target size, weak boundary information, low target–background contrast, frequent occlusion, and complex fruit-cluster backgrounds. Built upon YOLOv12s, ELCD-YOLO employs a heterogeneous dual-backbone architecture combining YOLOv12s and EfficientNetV2-N to extract complementary semantic and local texture representations from the same input image and integrate them through cross-backbone feature fusion.

To enhance small-object feature representation, the cross-layer reconstruction concept adopted from LSOD-YOLO was adapted to the heterogeneous dual-backbone fusion framework. The fused P3, P4, and P5 representations are reorganized into higher-resolution P2, P3, and P4 detection features without increasing the number of detection heads. In addition, CAFM is serially placed after the original A2C2 block to provide complementary convolution–attention feature refinement, while DySample is employed in the corresponding cross-scale reconstruction pathways for content-adaptive spatial alignment before feature fusion. The main contribution of ELCD-YOLO therefore lies in the task-oriented adaptation and integration of established architectural mechanisms, rather than in the development of fundamentally new feature-extraction, attention, or upsampling principles. This application-oriented configuration combines heterogeneous feature extraction, cross-backbone fusion, high-resolution cross-layer reconstruction, contextual–local feature refinement, and dynamic spatial alignment within a unified detection framework. The effectiveness of the overall configuration is evaluated experimentally, whereas potential interaction effects or synergistic gains among individual components have not been independently established.

Experiments were conducted using a grape-stem detection dataset composed predominantly of images acquired from the parallel fruit-sorting robot platform, together with a small number of publicly sourced grape images included only in the test subset. The original images were partitioned before augmentation, augmentation was restricted to the training subset, and identified acquisition-group overlaps were removed from the validation and test subsets. Under the final evaluation protocol, ELCD-YOLO achieved an mAP@0.5 of 68.00 $\pm$ 0.25%, a Precision of 92.51 $\pm$ 0.25%, and a Recall of 59.50 $\pm$ 0.35% across three independent training runs. Compared with the YOLOv12s baseline, these values correspond to improvements of 2.89, 3.10, and 4.18 percentage points, respectively. ELCD-YOLO required 21.5 GFLOPs, had a model size of 23.6 MB, and achieved a detector-pipeline inference speed of 81 FPS on an NVIDIA GeForce RTX 4070 GPU. Although the model size increased and the inference speed decreased relative to YOLOv12s, the reported computational complexity remained unchanged at 21.5 GFLOPs. The comparative and sequential ablation experiments support the effectiveness of the complete architectural configuration relative to the YOLOv12s baseline under the investigated conditions, while comparisons with other detection architectures reveal different trade-offs among detection accuracy, computational complexity, model size, and inference efficiency.

Despite these encouraging results, the generalizability of ELCD-YOLO remains subject to several limitations. The evaluation was conducted on a relatively limited, task-specific dataset, primarily collected using a single experimental robotic platform. Although a small number of publicly sourced grape images were included in the final test set, the evaluation does not constitute a systematic cross-dataset or cross-environment validation. Consequently, the generalizability of the proposed framework to substantially different grape varieties, illumination conditions, acquisition environments, and other low-contrast small-

object detection tasks has not yet been established. The reported performance improvements should therefore be interpreted within the scope of the present dataset, model configurations, and evaluation protocol. Broader cross-dataset and cross-environment experiments are required to further assess the robustness and applicability of the proposed framework.

Furthermore, the experimental validation in this study was limited to grape-stem detection and image-space bounding-box localization. The reported accuracy and inference-speed results therefore characterize the visual perception component rather than the operational performance of a complete robotic fruit-sorting system. Robot-coordinate localization accuracy, grasping success rate, sorting success rate, and complete robotic cycle time were not quantitatively evaluated. Accordingly, the present results support further investigation of ELCD-YOLO as a visual perception component for parallel fruit-sorting applications, while its effectiveness in complete closed-loop robotic manipulation remains to be established through dedicated robotic experiments.

Future work will focus on broader cross-dataset and cross-environment evaluations involving different grape varieties, illumination conditions, occlusion levels, and acquisition environments, as well as an assessment of transferability to other low-contrast small-object detection tasks. Model compression and deployment on edge-computing platforms will also be explored to further reduce storage and computational requirements. In addition, the proposed visual perception framework will be integrated with camera calibration, three-dimensional target localization, coordinate transformation, robot motion planning, and grasping control to enable quantitative evaluation of complete robotic sorting performance, including localization accuracy, grasping success rate, sorting efficiency, cycle time, and operational reliability.

Supplementary Materials: The following supporting information can be downloaded at: https://www.mdpi.com/article/10.3390/app162110694/s1, Table S1: Dataset composition and summary of acquisition-overlap screening; Table S2: Individual results of three independent training runs for the baseline and principal ablation configurations; Table S3: Stage-wise configuration of the ELCD-YOLO network architecture.

Author Contributions: Conceptualization, J.S. and G.G.; methodology, J.S. and G.G.; software, J.S.; validation, J.S.; formal analysis, J.S.; investigation, J.S.; resources, J.S. and G.G.; data curation, J.S.; writing—original draft preparation, J.S.; writing—review and editing, J.S. and G.G.; visualization, J.S.; supervision, G.G.; project administration, G.G. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The dataset generated and analyzed in this study is not publicly available because it forms part of ongoing research. Requests for access to the data may be directed to the corresponding author and will be considered on a case-by-case basis, subject to applicable research and data-sharing restrictions.

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

References

1. Albelwi, S.; Mahmood, A. A Framework for Designing the Architectures of Deep Convolutional Neural Networks. *Entropy* **2017**, *19*, 242. [CrossRef]
2. Mo, J.; Shao, Z.-F.; Guan, L.; Xie, F.; Tang, X. Dynamic Performance Analysis of the X4 High-Speed Pick-and-Place Parallel Robot. *Robot. Comput.-Integr. Manuf.* **2017**, *46*, 48–57. [CrossRef]
3. Sun, F.C.; Liu, C.F.; Huang, W.B.; Zhang, J.W. Object Classification and Grasp Planning Using Visual and Tactile Sensing. *IEEE Trans. Syst. Man Cybern. Syst.* **2017**, *46*, 48–57. [CrossRef]
4. Blanes, C.; Mellado, M.; Ortiz, C.; Valera, A. Technologies for Robot Grippers in Pick and Place Operations for Fresh Fruits and Vegetables. *Span. J. Agric. Res.* **2011**, *9*, 1130–1141. [CrossRef]

5. Przerwa, K.; Kasprzak, W.; Stefańczyk, M. Infrared Image-Based 3D Surface Reconstruction of Free-Form Texture-Less Objects. In Proceedings of the International Conference on Computer Recognition Systems (CORES), Wroclaw, Poland, 25–27 May 2015; pp. 809–820.
6. Yuan, T.; Xu, C.-G.; Ren, Y.-X.; Feng, Q.-C.; Tan, Y.-Z.; Li, W. Detecting the Information of Cucumber in Greenhouse for Picking Based on NIR Image. *Spectrosc. Spectr. Anal.* **2009**, *29*, 2054–2058.
7. Xing, J.; Jancsók, P.; De Baerdemaeker, J. Stem-End/Calyx Identification on Apples Using Contour Analysis in Multispectral Images. *Biosyst. Eng.* **2007**, *96*, 231–237. [CrossRef]
8. Eizentals, P.; Oka, K. 3D Pose Estimation of Green Pepper Fruit for Automated Harvesting. *Comput. Electron. Agric.* **2016**, *128*, 127–140. [CrossRef]
9. Qiao, J.; Sasao, A.; Shibusawa, S.; Kondo, N. Extracting External Features of Sweet Peppers Using Machine Vision System on Mobile Fruits Grading Robot. *Int. J. Food Eng.* **2012**, *8*, 1039–1058. [CrossRef]
10. Seema; Kumar, A.; Gill, G.S. Automatic Fruit Grading and Classification System Using Computer Vision: A Review. In Proceedings of the IEEE International Conference on Advances in Computing, Communications and Informatics (ICACCI), Kochi, India, 10–13 August 2015; pp. 598–603.
11. Lenz, I.; Lee, H.; Saxena, A. Deep Learning for Detecting Robotic Grasps. *Int. J. Robot. Res.* **2015**, *34*, 705–724. [CrossRef]
12. Lawal, M.O. Tomato Detection Based on Modified YOLOv3 Framework. *Sci. Rep.* **2021**, *11*, 1447. [CrossRef] [PubMed]
13. Chen, W.; Lu, S.; Liu, B.; Li, G.; Qian, T.; Huang, J. Detecting Citrus in Orchard Environment by Using Improved YOLOv4. *Sci. Program.* **2020**, *2020*, 8859237. [CrossRef]
14. Zhang, Q.; Gao, G.Q. Parallel Robot String Fruit Grasping Model and Grasping Pose Calculation. *Trans. Chin. Soc. Agric. Eng.* **2019**, *35*, 35–44.
15. Ning, Z.T.; Luo, L.F.; Liao, J.X.; Wen, H.J.; Wei, H.L.; Lu, Q.H. Grape Stem Recognition and Optimal Picking Positioning Based on Deep Learning. *Trans. Chin. Soc. Agric. Eng.* **2021**, *37*, 222–229. [CrossRef]
16. Girshick, R.; Donahue, J.; Darrell, T.; Malik, J. Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Columbus, OH, USA, 23–28 June 2014; pp. 570–578.
17. Girshick, R. Fast R-CNN. In Proceedings of the IEEE International Conference on Computer Vision (ICCV), Santiago, Chile, 7–13 December 2015; pp. 1440–1448.
18. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 1137–1149. [CrossRef] [PubMed]
19. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.-Y.; Berg, A.C. SSD: Single Shot Multibox Detector. In Proceedings of the European Conference on Computer Vision (ECCV), Amsterdam, The Netherlands, 11–14 October 2016; pp. 21–37.
20. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You Only Look Once: Unified, Real-Time Object Detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 779–788.
21. Redmon, J.; Farhadi, A. YOLO9000: Better, Faster, Stronger. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 7263–7271.
22. Redmon, J.; Farhadi, A. YOLOv3: An Incremental Improvement. *arXiv* **2018**, arXiv:1804.02767.
23. Tian, Y.; Ye, Q.; Doermann, D. YOLOv12: Attention-Centric Real-Time Object Detectors. *arXiv* **2025**, arXiv:2502.12524.
24. Tan, M.; Le, Q. EfficientNetV2: Smaller Models and Faster Training. In Proceedings of the 38th International Conference on Machine Learning (ICML), Virtual Event, 18–24 July 2021; Volume 139, pp. 10096–10106. Available online: https://proceedings.mlr.press/v139/tan21a.html (accessed on 17 July 2026).
25. Wang, H.; Liu, J.; Zhao, J.; Zhang, J.; Zhao, D. Precision and speed: LSOD-YOLO for lightweight small object detection. *Expert Syst. Appl.* **2025**, *269*, 126440. [CrossRef]
26. Liu, W.; Lu, H.; Fu, H.; Cao, Z. Learning to Upsample by Learning to Sample. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), Paris, France, 2–6 October 2023; pp. 6027–6037. [CrossRef]
27. Hu, S.; Gao, F.; Zhou, X.; Dong, J.; Du, Q. Hybrid Convolutional and Attention Network for Hyperspectral Image Denoising. *arXiv* **2024**, arXiv:2403.10067.
28. Zhao, X.Y.; Fang, J.D.; Zhao, Y.D. Improved Tomato Seedling Classification and Recognition Model of YOLOv5s. *Sci. Technol. Eng.* **2024**, *24*, 11774–11785.
29. Sun, J.; Wu, Z.Q.; Jia, Y.L.; Gong, D.J.; Wu, X.H.; Shen, J.F. Detecting Grape in an Orchard Using Improved YOLOv5s. *Trans. Chin. Soc. Agric. Eng.* **2023**, *39*, 192–200. [CrossRef]
30. Hao, J.J.; Bing, Z.K.; Yang, S.H.; Yang, J.; Sun, L. Improved YOLOv3 Algorithm for Detection of Green Walnut. *Trans. Chin. Soc. Agric. Eng.* **2022**, *38*, 183–190.
31. Lyu, S.; Li, R.; Zhao, Y.; Li, Z.; Fan, R.; Liu, S. Green Citrus Detection and Counting in Orchards Based on YOLOv5-CS and AI Edge System. *Sensors* **2022**, *22*, 576. [CrossRef] [PubMed]

32. Zhang, Y.; Zhou, B.; Zhao, X.; Song, X. Enhanced Object Detection in Low-Visibility Haze Conditions with YOLOv9s. *PLoS ONE* **2025**, *20*, e0317852. [CrossRef]
33. Lu, L.G.; He, D.B.; Deng, Z.; Liu, C.; Shen, Z. MASF-YOLO: An Improved YOLOv11 Network for Small Object Detection on Drone View. In Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics (SMC), Bratislava, Slovakia, 5–8 October 2025; pp. 2703–2708.
34. Ji, C.L.; Yu, T.; Gao, P.; Wang, F.; Yuan, R.-Y. Yolo-tla: An Efficient and Lightweight Small Object Detection Model Based on YOLOv5. *J. Real-Time Image Process.* **2024**, *21*, 141. [CrossRef]
35. Liu, Y. DiffYOLO: Object Detection for Anti-Noise via YOLO and Diffusion Models. *arXiv* **2024**, arXiv:2401.01659.
36. Wang, B.; Li, H.; Wei, L.; Zhang, Y. CSF-Net: Cross-Stage Fusion Network with Dual Backbone for Small Object Detection. *Sensors* **2026**, *26*, 1387. [CrossRef] [PubMed]
37. Zhang, Q.; Gao, G.Q. Prioritizing Robotic Grasping of Stacked Fruit Clusters Based on Stalk Location in RGB-D Images. *Comput. Electron. Agric.* **2020**, *172*, 105374. [CrossRef]
38. Zhang, Q.; Gao, G.Q. Hand-Eye Calibration and Grasping Pose Calculation with Motion Error Compensation and Vertical-Component Correction for 4-R(2-SS) Parallel Robot. *Int. J. Adv. Robot. Syst.* **2020**, *17*, 172988142090901. [CrossRef]

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