processes

MDPI

Article

BlueberryNet: A Lightweight CNN for Real-Time Ripeness Detection in Automated Blueberry Processing Systems

Bojian Yu [1], Hongwei Zhao [2] and Xinwei Zhang [3,*]

[1] School of Marine Engineering Equipment, Zhejiang Ocean University, Zhoushan 316022, China; 224473449@st.usst.edu.cn
[2] Qingdao Institute of Software, China University of Petroleum (East China), Qingdao 266580, China; zhw@s.upc.edu.cn
[3] School of Foreign Languages, Zhejiang Ocean University, Zhoushan 316022, China
* Correspondence: 2022115@zjou.edu.cn

Abstract

Blueberries are valued for their flavor and health benefits, but inconsistent ripeness at harvest complicates post-harvest food processing such as sorting and quality control. To address this, we propose a lightweight convolutional neural network (CNN) to detect blueberry ripeness in complex field environments, supporting efficient and automated food processing workflows. To meet the low-power and low-resource demands of embedded systems used in smart processing lines, we introduce a Grouped Large Kernel Reparameterization (GLKRep) module. This design reduces computational cost while enhancing the model's ability to recognize ripe blueberries under complex lighting and background conditions. We also propose a Unified Adaptive Multi-Scale Fusion (UMSF) detection head that adaptively integrates multi-scale features using a dynamic receptive field. This enables the model to detect blueberries of various sizes accurately, a common challenge in real-world harvests. During training, a Semantics-Aware IoU (SAIoU) loss function is used to improve the alignment between predicted and ground truth regions by emphasizing semantic consistency. The model achieves 98.1% accuracy with only 2.6M parameters, outperforming existing methods. Its high accuracy, compact size, and low computational load make it suitable for real-time deployment in embedded sorting and grading systems, bridging field detection and downstream food-processing tasks.

Keywords: fruit-ripeness detection; lightweight detection model; multi-scale feature extraction; lightweight convolutional neural network

check for updates

Academic Editor: John Anthony Rossiter

Received: 10 July 2025
Revised: 2 August 2025
Accepted: 7 August 2025
Published: 10 August 2025

Citation: Yu, B.; Zhao, H.; Zhang, X. BlueberryNet: A Lightweight CNN for Real-Time Ripeness Detection in Automated Blueberry Processing Systems. *Processes* **2025**, *13*, 2518. https://doi.org/10.3390/pr13082518

Copyright: © 2025 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

In the era of precision agriculture, the demand for intelligent harvesting technologies has grown rapidly due to increasing labor shortages, rising production costs, and the urgent need for quality and efficiency in fruit production [1,2]. Among the core tasks enabling automated harvesting, real-time and accurate fruit-ripeness detection plays a pivotal role in determining optimal picking times, minimizing post-harvest losses, and improving yield quality [3]. Traditional manual harvesting methods are often labor intensive, inconsistent, and unsuitable for large-scale orchards [4,5]. As a result, vision-based ripeness-detection systems have emerged as promising solutions for bridging the gap between traditional practices and intelligent agricultural operations [6,7].

Blueberries, characterized by their small size, clustered distribution, and rapid ripening cycle, are both an economically valuable crop and a highly challenging target for

intelligent harvesting [8–10]. Their nutritional richness-high in anthocyanins, vitamin C, and antioxidants has fueled global consumption and expansion of cultivation areas [11,12]. However, the fruits' tendency to rot rapidly after ripening and their physical complexity make detection and harvesting particularly difficult in natural orchard environments [13]. Overlapping fruits, occlusion from leaves and stems, and varying lighting conditions further exacerbate the detection challenge, especially for systems deployed on low-power embedded devices with limited computational capacity [14]. However, traditional machine-learning methods, such as color-based segmentation and shape-based analysis, often suffer from low detection accuracy, poor robustness, high computational cost, and slow processing speed, limiting their applicability in complex agricultural environments.

To address these issues, recent studies have applied improved object-detection models to the task of berry ripeness recognition. In particular, two-stage detection frameworks have significantly improved detection performance. Among them, the R-CNN family stands out as a representative approach. By introducing the Region Proposal Network (RPN), these models generate candidate object regions within an image and perform classification and localization through separate subnetworks, enabling accurate and robust object detection. In contrast, one-stage detection methods, represented by the YOLO series, remove the need for an RPN and directly perform object classification and bounding box regression in a single detection step, significantly improving detection speed while maintaining competitive accuracy. In [15], Chen et al. proposed the MTD-YOLOv7 model for cherry tomato bunch ripeness detection. By extending the YOLOv7 architecture with multi-task decoders, it simultaneously identifies fruit bunches, individual fruit ripeness, and bunch-level ripening. The model demonstrated strong robustness in complex agricultural environments, achieving 86.6% accuracy, making it promising for robotic harvesting applications. In [16], Zhu et al. developed YOLO-LM, a lightweight detector specifically for Camellia oleifera fruit in orchards. Incorporating Criss-Cross Attention (CCA) and Adaptive Spatial Feature Fusion (ASFF), the model improved detection accuracy in occluded environments (93.18% mAP@0.5) and facilitated orchard yield estimation and autonomous harvesters. In [17], Li et al. introduced a multi-view imaging-based phenotyping system (MARS-PhenoBot), integrating the Segment Anything Model (SAM) for label-free annotation, along with a customized BerryNet model. This system automates the detection of metrics such as fruit count, ripeness level, and cluster compactness, enabling high-throughput phenotyping in field conditions for precision breeding and management. In [18], Yang et al. developed an enhanced detail feature module (EDFM) with content-aware reassembly (CARAFE), improving feature extraction for color and texture, thus enhancing detection accuracy. In [19], Quiroz et al. validated a CNN-based model for identifying legacy blueberry growth stages in Chilean smart farms, demonstrating the versatility of deep learning in agricultural settings. Despite these advances, existing methods still struggle in visually complex scenes, particularly with occlusion and background interference. Moreover, high computational demands hinder their applicability to real-time embedded systems.

To overcome these challenges, we propose BlueberryNet, a novel lightweight and robust deep-learning framework tailored for high-accuracy blueberry ripeness detection in real-world orchard settings. Our approach is guided by three core intuitions: (1) accurate detection under occlusion requires strong global semantic representation; (2) multi-scale feature fusion should be dynamically adaptable to account for variability in fruit size and viewpoint; (3) loss functions must be sensitive to IoU quality and class imbalance in dense scenes.

To this end, BlueberryNet introduces three novel modules that jointly enhance accuracy, adaptability, and efficiency. First, the GLKRep module improves global semantic perception by leveraging reparameterized large-kernel convolutions, enabling wide receptive fields without increasing inference overhead. Second, the UMSF detection head dynamically fuses multi-scale features through learnable receptive field selection, enhancing robustness to varying fruit sizes and perspectives. Finally, the model incorporates the SAIoU loss function, which introduces semantic consistency constraints among regional features during regression optimization, thereby mitigating false detections under occlusion and class imbalance.

In contrast to previous lightweight detectors such as YOLOv5n and YOLOv8n—which rely on static feature fusion structures and conventional classification losses, BlueberryNet introduces both structural adaptability and sample-aware optimization into a compact architecture. This enables superior performance in complex orchard environments characterized by dense clustering, occlusion, and variable illumination. By jointly addressing semantic representation, multi-scale fusion, and supervisory quality, BlueberryNet achieves a favorable trade-off between accuracy and efficiency, making it a practical and deployable solution for real-time fruit-ripeness detection on edge devices.

The main contributions of this paper are summarized as follows:

(1) We construct a novel Grouped Large Kernel Reparameterization (GLKRep) module, which improves semantic representation using structurally reparameterized grouped convolutions, allowing for large receptive fields without increased inference cost.
(2) We propose an adaptive Unified Adaptive Multi-Scale Fusion Module (UMSF) detection head that dynamically fuses multi-scale features through learned receptive field selection, overcoming the rigidity of traditional FPN or PANet-based fusion used in YOLO-family models.
(3) We integrate Semantics-Aware IoU (SAIoU) Loss, which introduces semantic consistency constraints among regional features during the regression optimization process, enabling a more comprehensive and precise evaluation of the alignment between predicted and ground truth bounding boxes.

The rest of this paper is organized as follows: Section 2 introduces the dataset and preprocessing methods; Section 3 presents the BlueberryNet model; Section 4 reports experimental results; Section 5 discusses the findings; Section 6 concludes the paper and outlines future work.

2. Materials

Dataset Collecting

This article takes the ripeness of blueberries as the research object, and the data are sourced from the blueberry planting base of Yiran Ecological Family Farm in Xiaosha, Dinghai District, Zhejiang Province. Blueberry photos were taken in a natural state using an iPhone 14 at different ripeness levels. The collection dates were from 9:00 a.m. to 5:00 p.m. According to the actual situation of blueberry picking robots in blueberry orchards and the advice of experts, the ripeness of blueberries was divided into three categories: immature, semi-mature, and fully mature. Considering the diversity of the data set and the model's generalization ability as shown in Figure 1, the collected images covered a variety of conditions, including forward lighting, backlighting, distant views, close-up views, leaf occlusion, and fruit overlap. After screening and statistical analysis, a total of 2056 JPG format blueberry images were collected, with a resolution of 3024 × 4032 pixels.

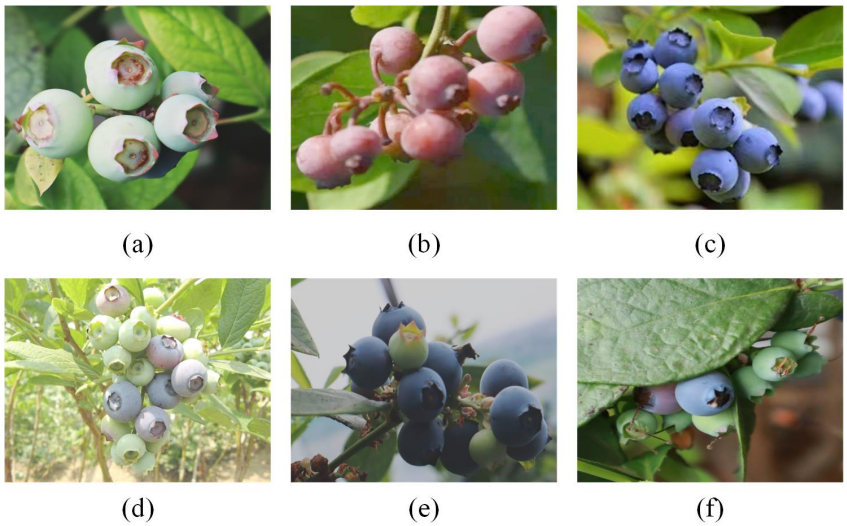


Figure 1. Example of blueberry fruit collection images. (**a**) Green fruit period; (**b**) purple fruit period; (**c**) mature fruit period; (**d**) back light; (**e**) bacing light; (**f**) occlusion.

3. Dataset Enhancement

To enhance the generalization ability and stability of the model, as well as to simulate various scenarios and noise, the experiment employed data-augmentation techniques to demonstrate the diversity and comprehensiveness of the dataset across different scenarios. As illustrated in Figure 2, vertical and horizontal flipping were employed to simulate shooting scenes from different angles and positions; Gaussian blur was utilized to eliminate noise and details from the images. The mosaic technique concatenates multiple images to simulate complex backgrounds in real environments and to identify occlusions between targets. In this experiment, the LabelImg software was employed to manually annotate the samples in the dataset. The maximum rectangular box was utilized to select the area of the blueberry fruit, generating a TXT file containing the annotation information upon completion. This file stores the filename of the fruit image, the coordinate information of the four corners of the annotation area, and the corresponding annotation category.

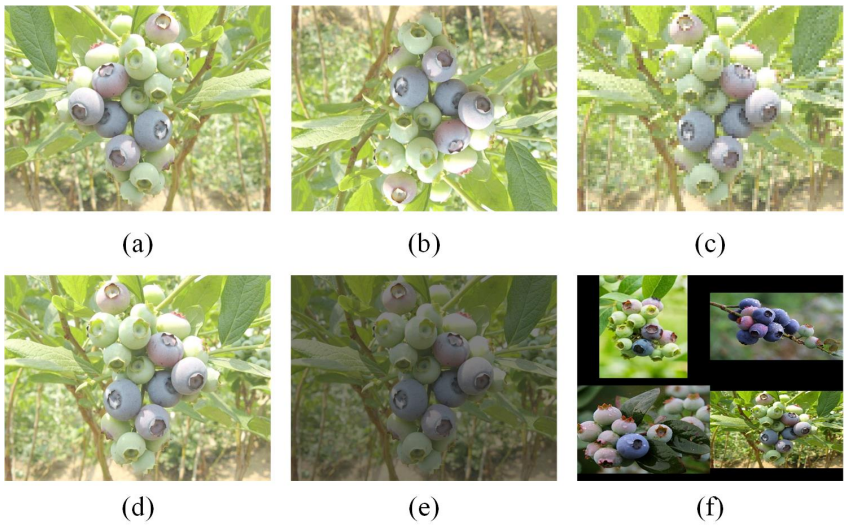


Figure 2. Samples after data enhancement. (**a**) Original image; (**b**) vertical flipping; (**c**) Gaussian blur; (**d**) horizontally flipping; (**e**) brightness weakening; (**f**) mosaic.

A total of 15,290 blueberry fruits were annotated in the dataset, and the images were resized to 640 × 640 pixels. To ensure the appropriate allocation of data and effective model training, the annotated dataset was randomly divided into training, validation, and testing sets in a ratio of 6:2:2. Table 1 shows the feature attribute table of the blueberry image dataset.

Table 1. Blueberry image dataset feature attributes.

Category	Attribute	Value/Description
Ripeness Classification	Total images	2000 images
	Green fruit period	6400 instances
	Purple fruit period	5650 instances
	Mature fruit period	3230 instances
	Back light	2195 instances
	Facing light	2760 instances
	Occlusion	1870 instances
	Total annotations	15,290 labeled fruits
Enhancement Classification	Original image	2050 images
	Vertical flipping	1070 images
	Gaussian blur	2630 images
	Horizontally flipping	1790 images
	Brightness weakening	2760 images
	Mosaic	900 images
	Total images	11,200 images include original image

4. Methods

4.1. Overview Structure

YOLOv8 provides multiple size variants, including n (nano), s (small), m (medium), l (large), and x (extra large), catering to different application requirements [20,21]. Among these, YOLOv8n is one of the most lightweight and fastest versions, making it suitable for resource-constrained environments such as embedded devices [22,23].

As illustrated in Figure 3, the BlueberryNet model adopts a Backbone–Neck–Head three-stage structure, aiming to improve performance in object-detection tasks. In the Backbone stage, the GLKRep module replaces the traditional SPPF layer. By combining channel mixing and group convolution mechanisms with structural reparameterization, it reduces computational complexity while enhancing the contextual awareness of feature extraction. The Neck section introduces a custom UMSF detection head, which can adaptively fuse feature maps from different levels, combined with size resampling and element-wise fusion, effectively improving the model's ability to adapt to objects of different scales. The Head section employs a three-scale parallel detection structure, with two CBS layers and one Conv layer, outputting both bounding box regression and object-classification results, enhancing the model's performance in multi-scale object detection. During training, BlueberryNet uses the SAIou Loss function, replacing the traditional YOLOv8 loss function. The SAIoU Loss integrates geometric location, center distance, and semantic content information, enabling the object-detection model to maintain strong robustness even under complex backgrounds, blurred boundaries, or slight occlusions. These innovative designs make BlueberryNet excel in object-detection tasks, especially improving detection accuracy in complex environments.

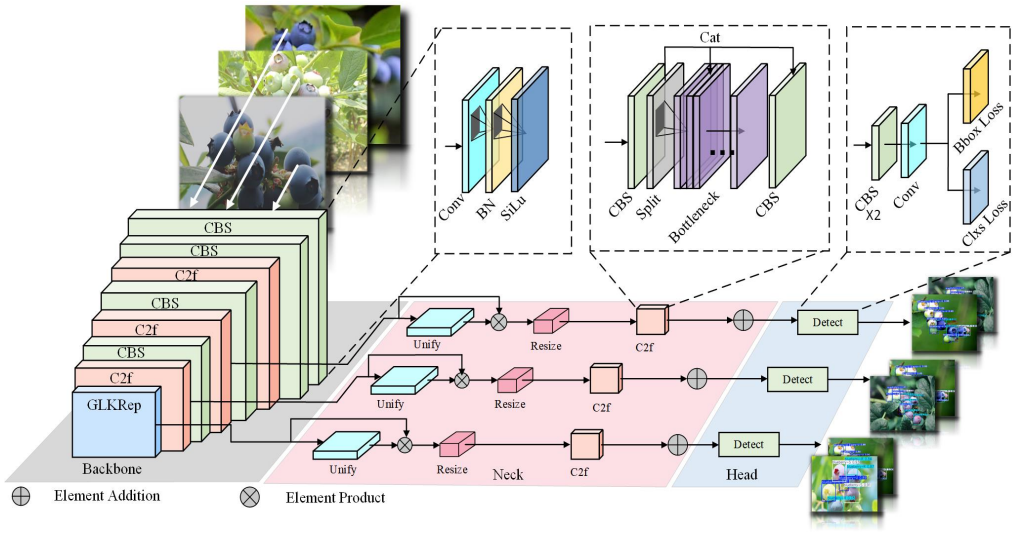


Figure 3. Structure of BlueberryNet.

4.2. GLKRep Modules

Real-time detection in complex field environments presents significant challenges when deploying large, computationally intensive models directly on edge devices [24]. To address the constraints posed by high computational demands and restricted resource availability, we propose a GLKRep module, designed for efficient semantic perception.

Our lightweight architecture aims to reduce model size and computational complexity while maintaining—or even improving—model performance. As illustrated in Figure 4, the GLKRep module employs channel-shuffled grouped convolutions combined with large kernel re-parameterization, optimizing both efficiency and accuracy for edge device deployment.

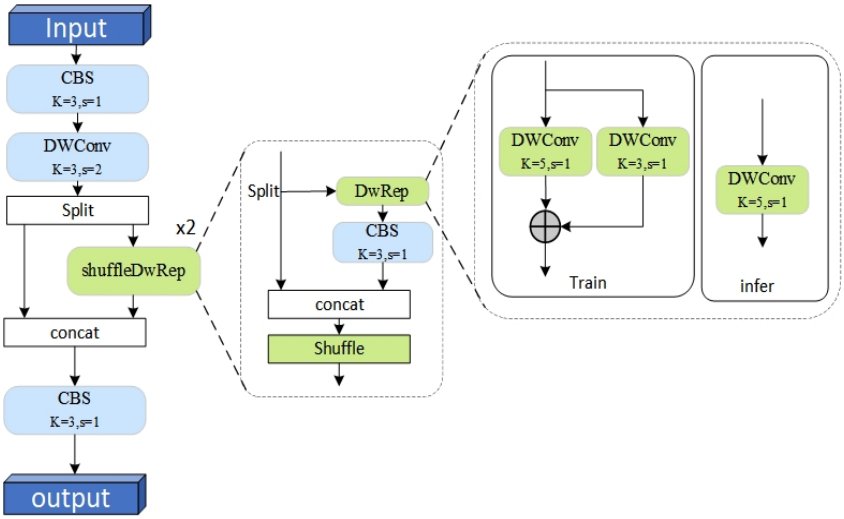


Figure 4. The architecture of the GLKRep module. It includes channel projection, depthwise downsampling, branch-wise reparameterization, and cross-channel shuffle-based fusion.

The module achieves a balance between computational efficiency and semantic perception capabilities through the following steps. The input feature map $F \in \mathbb{R}^{H \times W \times C_{in}}$ first undergoes a 1×1 convolution to reduce dimensionality, yielding F_{proj}. A depthwise separable convolution with stride 2 is then applied to perform spatial downsampling:

$$F_{dw} = DWConv_{3\times3,s=2}(F_{proj}). \quad (1)$$

The downsampled features are split into two branches. One branch passes through a depthwise reparameterized convolution (DWRep) followed by a 1×1 convolution for feature fusion:

$$F_{rep} = Conv_{1\times1}(DWRep(F_{split1})). \tag{2}$$

The other branch is retained and concatenated with F_{rep}, after which a channel shuffle operation is applied to promote cross-group interaction:

$$F_{shuffled} = Shuffle(F_{split2} \| F_{rep}). \tag{3}$$

Finally, the fused output is concatenated with the original projected features and passed through a 1×1 convolution to generate the output:

$$F_{out} = Conv_{1\times1}(F_{shuffled} \| F_{proj}). \tag{4}$$

By combining large-kernel receptive fields with grouped convolutions and structural reparameterization, the GLKRep module achieves an effective balance between semantic richness and computational efficiency. This enables enhanced global context modeling without introducing inference-time overhead. Moreover, its ability to capture fine-grained features under varying spatial scales and occlusions makes it particularly well-suited for complex orchard environments.

4.3. UMSF Modules

To address the computational inefficiency and limited generalization capability associated with redundant connections in traditional PANet-based neck architectures [25], we propose a dual-layer detection module that incorporates a receptive field adaptive mechanism. As illustrated in Figure 5, this module enhances the effectiveness and efficiency of feature representation through spatial alignment, dimensional unification, and dynamic multi-scale feature selection.

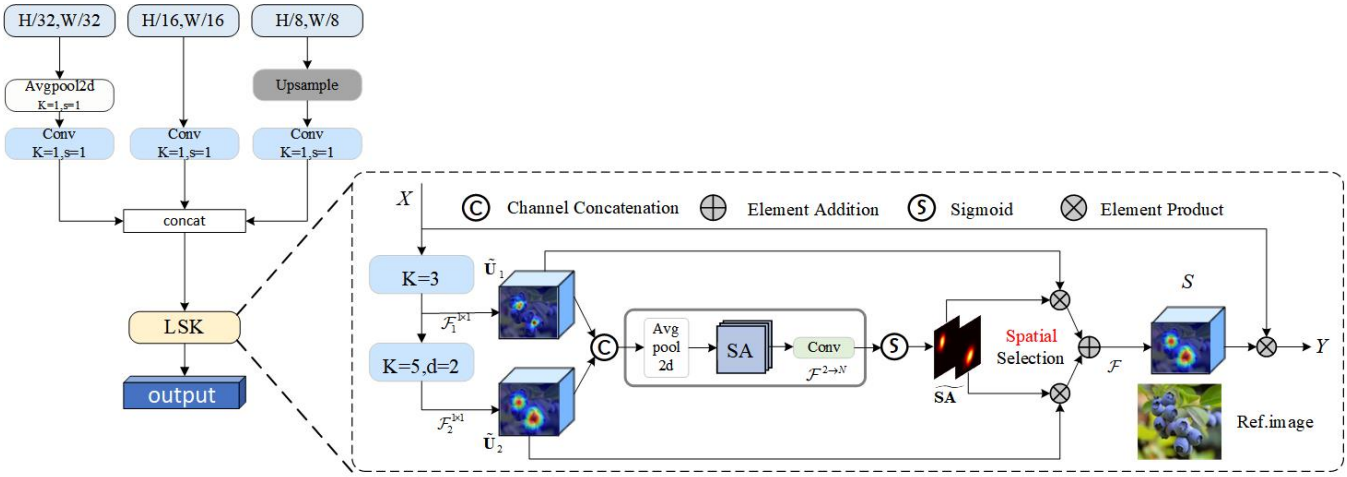


Figure 5. Architecture of the UMSF detection head module. The dual-branch design with dynamic attention-based fusion enables adaptive receptive field selection for robust multi-scale feature aggregation.

Specifically, the module first applies spatial normalization on multi-level input features through average pooling and bilinear upsampling to ensure consistent spatial dimensions. A 1×1 convolution is then used on the channel dimensions across different feature maps. These aligned and dimension-consistent features are concatenated and processed by a Lightweight Selective Kernel (LSK) mechanism, adapted from LSKNet.

The LSK mechanism stacks parallel 3×3 standard convolutions (to extract local details) and 5×5 dilated convolutions with a dilation rate of 2 (to capture global context). A learnable attention gate determines the importance of each receptive field branch and adaptively fuses them based on spatial content.

Finally, the fused features are passed through a feature pyramid re-parameterization strategy, enabling cascaded scale transformation and resize to form a dual-head detection structure. This further reduces computational redundancy and enhances inference speed, achieving a synergistic optimization of detection accuracy, scale robustness, and efficiency in complex orchard environments.

4.4. SAIou-Loss Function

To further improve the localization accuracy of bounding boxes in complex environments, this chapter proposes a novel loss function, the SAIoU-Loss function. In blueberry detection, the presence of densely packed targets and occlusions from leaves introduces significant challenges. Moreover, blueberries themselves often appear in dense clusters. Unlike traditional loss functions such as IoU [26], DIoU [27], or CIoU [28], which focus solely on the geometric overlap between bounding boxes, the SAIoU Loss incorporates semantic consistency constraints among regional features during regression optimization. This enables a more comprehensive evaluation of the alignment between predicted and ground truth boxes.

Let B_p and B_t denote the predicted and target bounding boxes, and let F_p and F_t be their corresponding semantic features extracted via RoIAlign [29] from the shared feature map. The SAIoU loss is defined as

$$\mathcal{L}_{\text{SAIoU}} = 1 - \left(\lambda \cdot \text{IoU}(B_p, B_t) + (1-\lambda) \cdot \text{Sim}(F_p, F_t)\right) + \delta \cdot \|c_p - c_t\|_2^2 \tag{5}$$

where $\text{IoU}(B_p, B_t)$ is the intersection-over-union between B_p and B_t. $\text{Sim}(F_p, F_t)$ is the cosine similarity between the semantic features:

$$\text{Sim}(F_p, F_t) = \frac{F_p \cdot F_t}{\|F_p\| \cdot \|F_t\|} \tag{6}$$

where c_p and c_t are the centers of B_p and B_t, respectively. $\lambda \in [0, 1]$ balances spatial and semantic contributions. δ is a weighting factor for the center distance penalty.

This design ensures that the model is not overly penalized for small localization errors if the predicted region captures the correct semantics. It encourages semantically aligned predictions, especially in challenging cases with occlusion, ambiguous boundaries, or visually similar classes.

5. Experiments

5.1. Experimental Details

To evaluate the performance, generalization, and efficiency of the proposed BlueberryNet model, we conducted a series of quantitative and qualitative experiments against multiple baselines.

The experimental models were trained, validated, and tested on a Windows 10 (64-bit) operating system. The computer used had 32 GB of RAM, an NVIDIA GeForce RTX 2060 GPU (NVIDIA Corporation, Santa Clara, CA, USA), and an Intel(R) Core(TM) i7-10870H CPU @ 2.20 GHz (Intel Corporation, Santa Clara, CA, USA). The PyTorch version was 1.10.0, the programming language was Python 3.8.5, and CUDA 11.3 was used for GPU acceleration.

All experiments in this study were conducted under identical conditions. The training images were resized to 640 × 640 pixels, with a batch size of 16. The initial learning rate was set to 0.01, and the optimizer used for training was SGD, with a momentum value of 0.937. The training process was carried out for a total of 120 epochs.

5.2. Evaluation Indicators

This study primarily evaluates model performance using precision (P), recall (R), mean average precision (mAP), floating point operations (FLOPs), frames per second (FPS), and the number of parameters (parameters, M),which are defined as follows [30,31]:

$$P = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}}, \tag{7}$$

where true positive (TP) represents the number of actual positive samples correctly predicted as positive, while false positive (FP) refers to the number of actual negative samples incorrectly predicted as positive.

$$R = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}}, \tag{8}$$

In the formula, false negative (FN) indicates the number of actual positive samples predicted as negative.

$$\mathrm{mAP} = \frac{1}{N} \sum_{i=1}^{N} \mathrm{AP}_i, \tag{9}$$

where average precision (AP) measures the average precision for a specific class of targets at various recall points and corresponds to the area under the precision–recall (PR) curve. When the Intersection over Union (IoU) threshold is set to 0.5, AP is specifically denoted as AP50.

$$\mathrm{FPS} = \frac{1}{T_{\mathrm{inference}}}, \tag{10}$$

where real-time performance is assessed using frames per second (FPS), where a higher FPS value indicates better real-time detection capability. These metrics collectively evaluate the accuracy and efficiency of the model in detecting blueberry ripeness.

5.3. Performance Comparison of BlueberryNet and YOLOv8n

This study proposes BlueberryNet by making structural improvements based on YOLOv8n. To verify the effectiveness of the improved model, a series of comparative experiments were conducted in which several representative test images were randomly selected for comparison.

As shown in Figure 6, The four sets of comparative images illustrate the following limitations of YOLOv8n when dealing with small, dense, and occluded blueberry targets. First, YOLOv8n exhibits a noticeable tendency for missed detections. In the first row, YOLOv8n identifies only a subset of the fruits, with the number of bounding boxes significantly lower than the actual number of blueberries—particularly for those located near the edges or partially occluded by leaves. In contrast, BlueberryNet accurately localizes most of the fruits in the scene. Second, YOLOv8n suffers from low localization precision and blurred bounding box boundaries. In the second row, multiple predicted boxes from YOLOv8n overlap substantially, which hinders target differentiation and negatively impacts subsequent tasks such as fruit counting and recognition. BlueberryNet, however, produces tighter and more precise bounding boxes that closely fit the fruit contours, demonstrating improved localization performance. Finally, YOLOv8n tends to produce redundant detections in densely packed scenarios. As shown in the fourth row, YOLOv8n outputs several overlapping boxes within the same region, leading to detection clustering. BlueberryNet effectively

mitigates this issue by incorporating a more efficient feature extraction mechanism, which enhances its ability to detect densely distributed small targets.

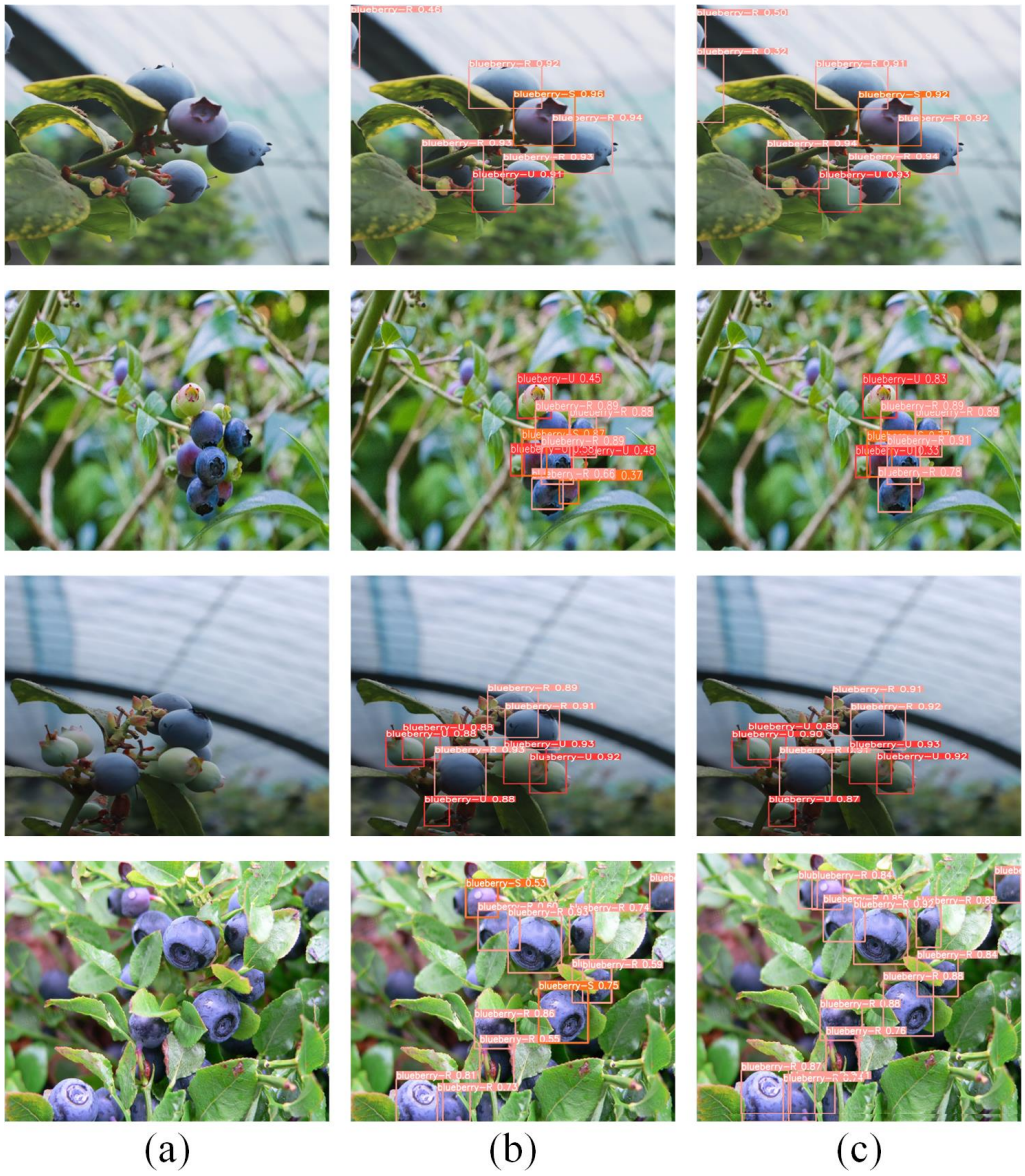


Figure 6. BlueberryNet detection experiment results: (**a**) original image; (**b**) YOLOv8n; (**c**) BlueberryNet (ours).

Figure 7 presents the performance curves of YOLOv8n and the improved BlueberryNet model during the training process.

An analysis of four key metrics reveals that BlueberryNet demonstrates superior learning capability and convergence speed from the early stages of training. In terms of recall, BlueberryNet shows rapid improvement within the first 20 epochs and maintains a stable value above 0.96% throughout the remainder of the training. This indicates stronger object detection capability and a lower risk of missed detections. For precision, BlueberryNet achieves a relatively stable curve with minimal fluctuations, reflecting robust training stability and better generalization. In contrast, YOLOv8n exhibits less stability in the early stages and maintains a comparatively lower precision overall. Regarding the mAP@0.5 metric, BlueberryNet sustains a high level of accuracy above 0.95 in the later stages of training, significantly outperforming YOLOv8n, which plateaus around

0.90. Overall, BlueberryNet consistently outperforms YOLOv8n across all key performance metrics—recall, precision, mAP@0.5, and mAP@0.5:0.95.

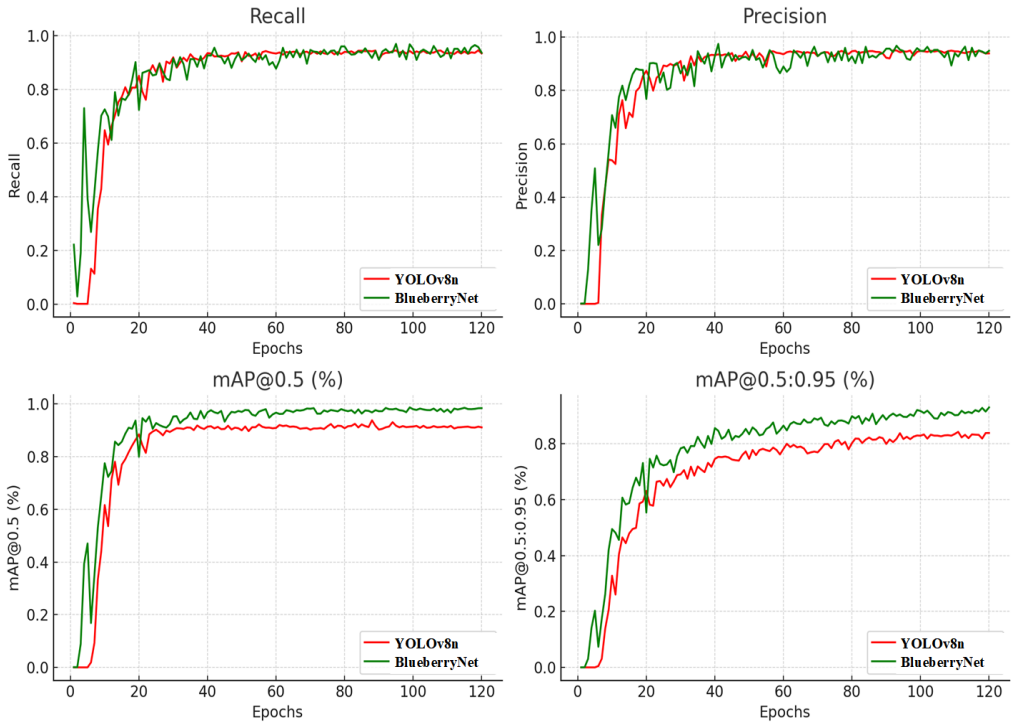


Figure 7. Comparison of the training curves of YOLOv8n and BlueberryNet.

Furthermore, its performance advantage becomes increasingly evident as training progresses, validating the effectiveness and superiority of the proposed model in blueberry-detection tasks.

5.4. Comparison Experiments

To evaluate the detection performance of the proposed BlueberryNet algorithm, a comprehensive comparison was conducted against several mainstream object-detection models, including Faster R-CNN [32], SSD [33], YOLOv5n [34], YOLOv7n-tiny [35], YOLOv8n [36], YOLOv9t [37], YOLOv10n [38], YOLOv11n [39], and BlueberryNet. All models were trained under identical environments and hyperparameter settings to ensure a fair evaluation. The performance comparison results are presented in Table 2.

Table 2. Comparison of model performance.

Models	Precision P/%	Recall R/%	mAP@0.5–0.95/%	FLOPs/GB	Parameters/M
Faster R-CNN	96.2	97.1	91.7	8.2	3.0
SDD	89.3	91.5	89.1	35.2	26.3
YOLOv5n	93.2	94.8	94.6	24.0	9.1
YOLOv7n-tiny	95.3	95.4	93.8	11.9	4.2
YOLOv8n	95.7	96.2	94.2	8.9	3.0
YOLOv9t	95.9	95.5	95.2	7.8	2.8
YOLOv10n	96.2	95.7	94.9	28.7	11.1
YOLOv11	96.7	93.9	82.1	13.2	6.0
BlueberryNet (Ours)	98.1	95.5	97.5	7.2	2.6

As shown in Table 2, BlueberryNet achieves the most outstanding performance, with a precision of 98.1%, a recall of 95.5%, and a mAP of 97.5%. Furthermore, it maintains a lightweight structure with only 2.6M parameters and a computational cost of 7.2 GFLOPs, making it highly suitable for deployment in resource-constrained environments. These results are largely attributed to the integration of the improved GLKRep module at the end of the backbone, which enhances contextual awareness while significantly reducing the number of parameters. Additionally, the UMSF detection head module is capable of aggregating multi-level feature maps and adaptively optimizing multi-scale feature fusion through a dynamic receptive field selection mechanism, thereby improving the model's ability to detect blueberry targets of varying sizes. In comparison, Faster R-CNN achieves a mAP@0.5–0.95% of 91.7%, precision of 96.2%, and recall of 97.1%. Although its recall is slightly higher, it comes at the cost of increased model complexity 3.0M parameters and computational load 8.2 GFLOPs, as typical of two-stage detection frameworks. SSD attains the highest recall among single-stage models but suffers from excessive complexity, resulting in low inference efficiency and poor suitability for lightweight applications. YOLOv10n and YOLOv11n generate overly large weight files, limiting their compatibility with embedded platforms. Although YOLOv5n, YOLOv7n-tiny, YOLOv8n, and YOLOv9t each offer different trade-offs in accuracy and efficiency, none achieves the optimal balance demonstrated by BlueberryNet.

The proposed model proves especially effective in dense, small-object scenarios such as real-time blueberry ripeness detection in complex orchard environments.

To provide an intuitive comparison of model performance, representative test images were selected, as shown in Figure 8.



Figure 8. Detection effect of different models.

In the first row of images, the blueberry fruits are densely packed with noticeable differences in ripeness, where light green and light purple fruits are interspersed among dark, mature fruits, creating some identification interference. In this scenario, Faster R-CNN can generally detect most dark fruits but fails to effectively identify lighter-colored or partially occluded fruits, with some bounding boxes erroneously offset to leaf areas, resulting in significant errors. YOLOv5n detects more targets, but the confidence scores are unevenly distributed, with severe overlapping of some bounding boxes and instances of

false detections and redundant boxes. YOLOv7-tiny shows improvement in detecting edge fruits but still misses some lighter-colored fruits. YOLOv10 and YOLOv11 produce more compact bounding box distributions and clearer boundary delineations among overlapping fruits, with YOLOv11 achieving relatively accurate localization of some partially mature fruits. BlueberryNet performs the best in this scenario, not only identifying all mature fruits but also accurately detecting two light green, unripe fruits, demonstrating that the model has learned the appearance features of fruits at different growth stages during training.

In the second row of images, the blueberry fruits are more sparsely distributed, but the background features significant leaf occlusion, posing a challenge to the models' anti-interference capabilities. Faster R-CNN only detects some foreground fruits, failing to penetrate leaf occlusion to identify targets in the background. YOLOv5n and YOLOv7-tiny show an increase in the number of detections but still miss several mature fruits. YOLOv10 and YOLOv11 effectively avoid misjudgments caused by leaf veins or reflections through accurate extraction of fruit edge contours, achieving significantly better detection performance than the previous models. BlueberryNet once again demonstrates precise recognition of small-scale and leaf-occluded fruits, even accurately boxing a fruit with only a partially exposed peel, showcasing significantly enhanced robustness and target perception capabilities.

The third row of images depicts blueberry fruits in a greenhouse environment with complex background structures and some degree of uneven lighting. Faster R-CNN's detection performance further declines, failing to mark most edge targets except for clearly visible foreground fruits. YOLOv5n and YOLOv7-tiny show slightly improved adaptability to the environment, but misdetections persist in areas with light spots or highly reflective leaves. YOLOv10 produces bounding boxes that better align with fruit contours, reducing the false detection rate, while YOLOv11 maintains boundary independence among multiple overlapping fruits. BlueberryNet again exhibits strong recognition capabilities in occluded and unevenly lit areas, particularly in the heavily occluded lower-left region of the image, where it successfully detects targets completely missed by other models, maintaining high confidence scores.

In the fourth row of images, the blueberries exhibit significant ripeness variations, ranging from light green, pink, purple, to dark blue fruits. Faster R-CNN and YOLOv5n almost entirely fail to identify non-dark fruits, resulting in low bounding box density and insufficient accuracy. YOLOv7-tiny responds to purple fruits to some extent but suffers from fragmented recognition and redundant boxes. YOLOv10 and YOLOv11 stably detect fruits of medium to high ripeness with balanced confidence score distributions. BlueberryNet comprehensively covers fruits of all color stages, achieving the highest detection count with almost no false positives, indicating that its training dataset likely includes blueberry samples under different spectra, endowing the model with superior spectral robustness and ripeness perception capabilities.

Figure 9 presents a radar chart comparing the performance metrics of BlueberryNet with those of other benchmark models. The figure provides a clear visual illustration of BlueberryNet's strengths, particularly in terms of its lightweight design. It achieves the highest scores in both parameter count (Params) and computational complexity (FLOPs), indicating its excellent suitability for deployment on resource-constrained devices.

Considering multiple aspects—including the number of detected objects, confidence score distribution, boundary localization accuracy, occlusion handling, and ripeness stage diversity—BlueberryNet consistently outperforms competing models. These results highlight its strong potential for real-world applications, especially in scenarios requiring high efficiency and robustness in complex agricultural environments.

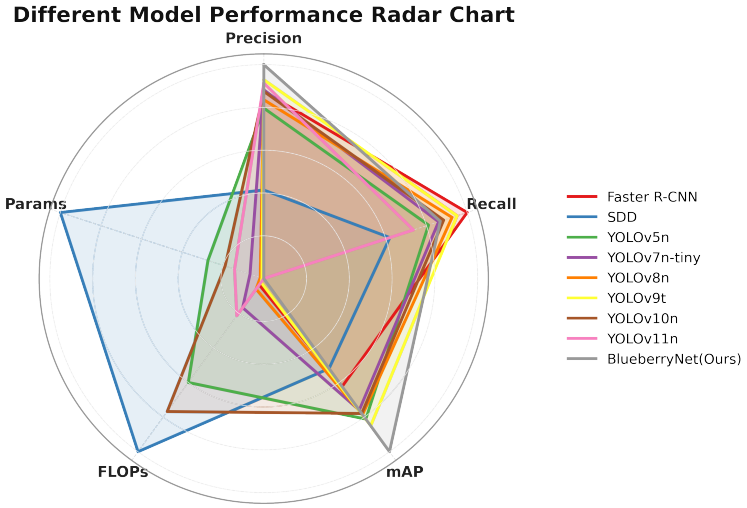


Figure 9. Different model performances radar chart.

5.5. Ablation Experiments

To evaluate the effectiveness of each proposed improvement, we conducted four ablation experiments under identical datasets and training settings. The experiments were carried out in a stepwise manner: first, the original SPPF layer in the backbone network was replaced with the custom-designed GLKRep module; second, the PANet structure in the neck network was substituted with the UMSF detection head; and finally, the original classification loss was replaced with the SAIou Loss function. The detailed results are presented in Table 3.

Table 3. Results of ablation experiments.

Baseline	+GLKRep	+UMSF	+SAIou	mAP@0.5–0.95/%	FLOPs/GB	Parameters/M
✓	×	×	×	91.7	8.2	3.0
✓	✓	×	×	94.1	7.9	2.8
✓	✓	×	✓	95.3	7.5	2.8
✓	✓	✓	×	96.8	7.5	2.7
✓	✓	✓	✓	98.1	7.2	2.6

The baseline model, without any modifications, achieved an mAP@0.5–0.95 of 91.7%, with 8.2 GFLOPs and 3.0M parameters. After introducing the GLKRep module, the mAP increased to 94.1%, while both FLOPs and parameter count decreased by approximately 0.2 units, indicating that the module enhances local feature extraction and receptive field representation without sacrificing computational efficiency.

Building upon this, replacing the PANet in the neck with the UMSF detection head module led to a further increase in mAP to 96.8%, and a reduction in model size to 2.7M parameters. This improvement is attributed to the UMSF detection head's ability to receive multi-level feature maps and adaptively optimize multi-scale feature fusion through a dynamic receptive field selection mechanism. These results further validate the module's effectiveness in enhancing detection performance for blueberry targets of varying sizes.

Finally, after integrating both structural improvements, we replaced the original loss function with the SAIou Loss. This led to a final mAP of 97.5%, with 7.2 GFLOPs and 2.6M parameters. The SAIou Loss improves the model's discriminative power by weighting positive and negative samples based on IoU-aware scores, thereby enhancing detection accuracy under occlusion and reducing false negatives and false positives. The results demonstrate a well-balanced improvement in both detection accuracy and model efficiency.

Figure 10 presents feature map visualizations to compare the performance of the GLKRep module and UMSF detection head during the feature extraction process. Figure 10a shows the original input image containing multiple blueberry fruits in a complex background. In Figure 10b, the feature maps extracted after incorporating the GLKRep module exhibit strong responses to object edges and local textures, effectively highlighting the structural information of blueberry fruits. This enhancement contributes to more precise localization by emphasizing fine-grained details. Figure 10c shows the feature maps after introducing the UMSF detection head. The results demonstrate improved semantic consistency and spatial continuity, with feature activations more concentrated in the blueberry regions and significantly reduced background interference. This indicates superior global feature fusion capability. Overall, the GLKRep module enhances the extraction of fine details, while the UMSF detection head module strengthens multi-scale semantic fusion. The combination of both modules substantially improves the model's robustness and accuracy in complex detection scenarios.

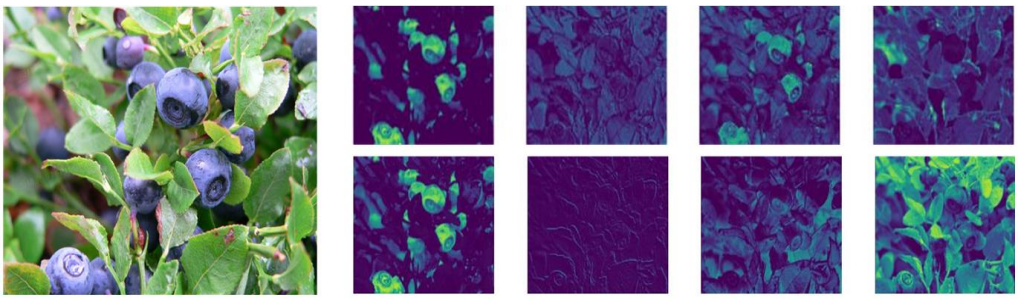


Figure 10. Visualization of feature maps: (**a**) original image; (**b**) GLKRep module output; (**c**) UMSF detection head output.

To intuitively illustrate the effectiveness of the proposed model improvements, we employ Gradient-weighted Class Activation Mapping (Grad-CAM) to visualize the attention regions of the detection targets [40,41]. In the visualizations, brighter areas indicate regions to which the model pays greater attention. The results are shown in Figure 11.

The first Figure 11a shows the original input images without any module. The second Figure 11b displays the heatmaps from the baseline YOLOv8n model. As observed, the attention is relatively scattered and inconsistent, with noticeable activation on non-target areas such as leaves and background structures. This reflects the model's limited ability to focus accurately on the blueberry regions in cluttered environments.

When the GLKRep module is incorporated (Figure 11c), the attention becomes significantly more concentrated on the actual blueberry targets. The model learns to emphasize relevant local features, thereby enhancing discrimination between foreground and background. However, some residual background interference remains under certain complex conditions.

With the integration of the UMSF detection head Figure 11d), the model gains improved global contextual awareness, enabling it to better aggregate multi-scale information and suppress irrelevant activations. The heatmaps show a more coherent and stable attention focus, even under occlusions or partial visibility of the fruit.

The final Figure 11e shows the results from the full model that combines both the GLKRep module and UMSF detection head. This configuration yields the most accurate and robust attention distribution, with clear, sharply localized focus on the blueberry regions and minimal response to background clutter.

These visual results confirm that the proposed architectural modifications contribute to improved feature representation and localization precision, particularly in complex field environments.

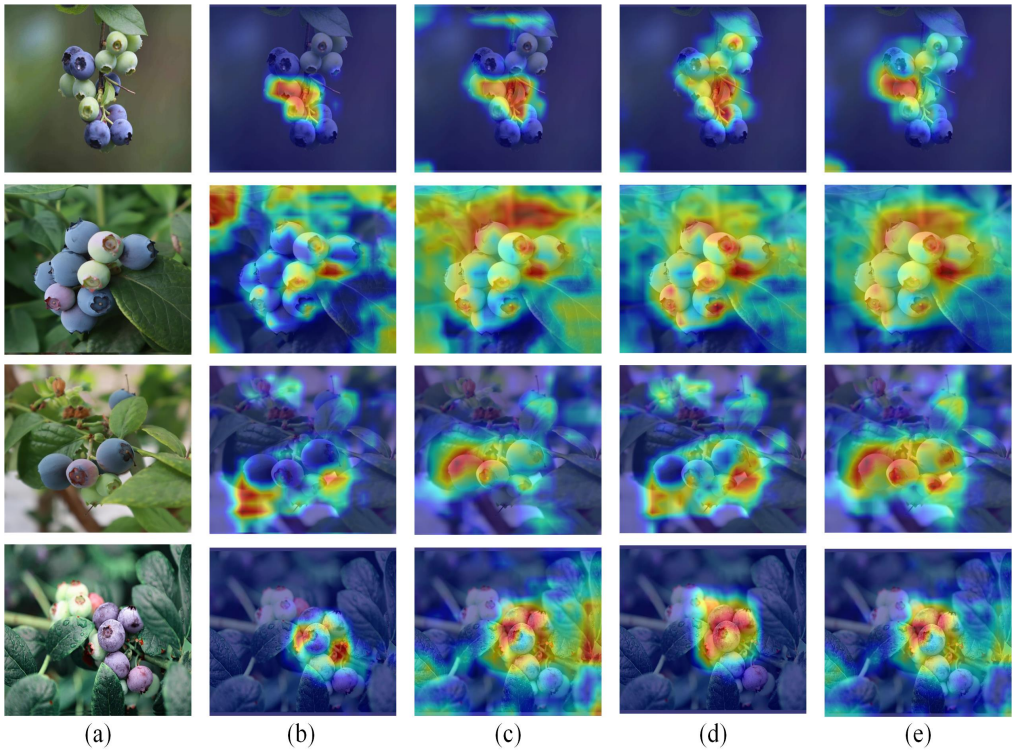


Figure 11. Visualization of heatmap. (**a**) Original image; (**b**) YOLOv8n; (**c**) YOLOv8n + GLKRep; (**d**) YOLOv8n + UMSF detection head; (**e**) YOLOv8n + GLKRep + UMSF.

5.6. Generalization Assessment

To evaluate the generalization capability of the BlueberryNet model, this study conducted tests on a publicly available blueberry dataset from the literature [42]. As shown in Figure 12, the left column presents the original images, while the right column displays the detection results produced by BlueberryNet. The experimental results demonstrate that the model can accurately detect blueberry fruits and distinguish their ripeness levels under complex background conditions. Even in scenarios involving dense fruit clusters, partial occlusion, or significant lighting variations, the model consistently produces precise bounding boxes and class labels, exhibiting strong robustness and high detection accuracy. These findings further confirm that BlueberryNet maintains reliable consistency and recognition performance across various shooting angles and application scenarios.



Figure 12. Model generalization detection diagram. (**a**) Original image; (**b**) BlueberryNet.

Table 4 compares the performance of YOLO-BLBE and BlueberryNet across multiple evaluation metrics. As shown in the table, BlueberryNet demonstrates higher efficiency in terms of model size and detection speed, while also outperforming YOLO-BLBE in key detection indicators such as precision, recall, and mAP. Although its F1 score is slightly lower than that of YOLO-BLBE, the overall performance of BlueberryNet is more balanced. These results indicate that BlueberryNet not only performs well on the specific training dataset but also exhibits strong detection capability and generalization on the public test set.

Table 4. Comparison of model performances.

Models	Model Size MB	Detection Speed s/per Image	Precision P/%	Recall R/%	F1 Score	mAP/%
YOLO-BLBE	12.75	0.009	93.72	97.56	95.60	98.14
BlueberryNet (Ours)	3.08	0.005	97.51	98.52	95.43	98.51

6. Conclusions

This paper proposes a lightweight and robust deep-learning framework, BlueberryNet, for high-accuracy blueberry ripeness detection.To achieve accurate identification of blueberries at different ripeness stages and address the limitations of existing detection models in multi-scale feature extraction and adaptability to complex environments, this study develops a lightweight BlueberryNet model based on the YOLOv8n architecture. The proposed model incorporates a GLKRep module to enhance semantic perception capability and introduces a UMSF detection head dual-layer detection module to adapt to multi-scale feature fusion requirements. The advantages of the proposed model are demonstrated in three key aspects:

(1) Lightweight design and deep integration of structural reparameterization: This paper innovatively introduces the GLKRep module, which combines grouped channel convolution with large-kernel structural reparameterization. This approach significantly reduces computational complexity while maintaining semantic perception capabilities, effectively enhancing the depth and semantic awareness of feature extraction. It ensures efficient deployment and real-time response on edge devices.

(2) Adaptive dual-layer receptive field multi-scale fusion structure: To address the significant scale variations and complex spatial distribution of blueberries in natural environments, a UMSF detection head dual-layer detection module was designed. This module dynamically receives and fuses feature maps from different layers of the backbone network, utilizing a multi-scale convolutional structure to achieve precise blueberry fruit recognition under conditions of scale variation, target overlap, and perspective changes, significantly enhancing the model's robustness in identifying blueberry ripeness in complex scenarios.

(3) Introduction of IoU-Aware classification loss to optimize detection consistency: During the model training phase, the SAIou Loss function is introduced, leveraging IoU-Aware Classification Scores (IACS) to effectively coordinate the optimization of target classification and bounding box regression tasks. This results in higher stability and accuracy in multi-target detection scenarios with dense fruit clusters and severe occlusion.

Despite the significant breakthroughs achieved by the BlueberryNet model in the blueberry ripeness-detection task, certain limitations remain. First, the model currently focuses on ripeness classification and does not address the detection of fruit pests or diseases. Second, the model relies on high-quality image inputs, and its adaptability to extreme weather conditions or blurry images needs further improvement. Future research will focus on enhancing the BlueberryNet model, further exploring its detection and recognition

capabilities for blueberry targets in complex agricultural scenarios. This includes achieving precise identification and classification of blueberry pests and diseases, intelligent estimation of large-scale blueberry yields, and analysis of blueberry growth trends, ultimately contributing to the promotion and development of intelligent agricultural monitoring technologies. In addition, we plan to deploy BlueberryNet on mobile and embedded platforms such as NVIDIA Jetson Nano, enabling real-time inference in orchards and post-harvest processing lines. These enhancements will further promote the deployment of AI-based monitoring systems in precision agriculture and facilitate the transition toward autonomous fruit production management.

Author Contributions: Conceptualization, B.Y.; methodology, H.Z.; resources, B.Y.; data curation, B.Y.; writing—original draft preparation, B.Y. project administration, H.Z. and X.Z. All authors contributed equally as co-first authors. All authors have read and agreed to the final version of the manuscript.

Funding: This research was funded by Zhejiang Ocean University, grant number JX6311040123.

Data Availability Statement: The annotated dataset and source code used in this study are publicly available at https://github.com/yubojian123/YOLO-Blueberry (accessed on 1 August 2025).

Conflicts of Interest: The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

1. Meshram, V.; Patil, K.; Meshram, V.; Hanchate, D.; Ramkteke, S. Machine learning in agriculture domain: A state-of-art survey. *Artif. Intell. Life Sci.* **2021**, *1*, 100010. [CrossRef]
2. Zhao, H.; Tang, Z.; Li, Z.; Dong, Y.; Si, Y.; Lu, M.; Panoutsos, G. Real-time object detection and robotic manipulation for agriculture using a YOLO-based learning approach. In Proceedings of the 2024 IEEE International Conference on Industrial Technology (ICIT), Bristol, UK, 25–27 March 2024; pp. 1–6.
3. Saleem, M.H.; Potgieter, J.; Arif, K.M. Automation in agriculture by machine and deep learning techniques: A review of recent developments. *Precis. Agric.* **2021**, *22*, 2053–2091. [CrossRef]
4. Javaid, M.; Haleem, A.; Khan, I.H.; Suman, R. Understanding the potential applications of Artificial Intelligence in Agriculture Sector. *Adv. Agrochem* **2023**, *2*, 15–30. [CrossRef]
5. Mana, A.; Allouhi, A.; Hamrani, A.; Rehman, S.; El Jamaoui, I.; Jayachandran, K. Sustainable AI-based production agriculture: Exploring AI applications and implications in agricultural practices. *Smart Agric. Technol.* **2024**, *7*, 100416. [CrossRef]
6. Albahar, M. A survey on deep learning and its impact on agriculture: Challenges and opportunities. *Agriculture* **2023**, *13*, 540. [CrossRef]
7. Shaikh, T.A.; Rasool, T.; Lone, F.R. Towards leveraging the role of machine learning and artificial intelligence in precision agriculture and smart farming. *Comput. Electron. Agric.* **2022**, *198*, 107119. [CrossRef]
8. Badgujar, C.M.; Poulose, A.; Gan, H. Agricultural object detection with You Only Look Once (YOLO) Algorithm: A bibliometric and systematic literature review. *Comput. Electron. Agric.* **2024**, *223*, 109090. [CrossRef]
9. Attri, I.; Awasthi, L.K.; Sharma, T.P.; Rathee, P. A review of deep learning techniques used in agriculture. *Ecol. Inform.* **2023**, *77*, 102217. [CrossRef]
10. Padhiary, M. The Convergence of Deep Learning, IoT, Sensors, and Farm Machinery in Agriculture. In *Designing Sustainable Internet of Things Solutions for Smart Industries*; IGI Global: Hershey, PA, USA, 2025; pp. 109–142.
11. Mandal, S.; Yadav, A.; Panme, F.A.; Devi, K.M.; SM, S.K. Adaption of smart applications in agriculture to enhance production. *Smart Agric. Technol.* **2024**, *7*, 100431. [CrossRef]
12. Duan, Y.; Tarafdar, A.; Chaurasia, D.; Singh, A.; Bhargava, P.C.; Yang, J.; Li, Z.; Ni, X.; Tian, Y.; Li, H.; et al. Blueberry fruit valorization and valuable constituents: A review. *Int. J. Food Microbiol.* **2022**, *381*, 109890. [CrossRef]
13. Wang, J.; Tian, J.; Li, D.; Gao, N.; Deng, J.; Yang, X.; Wang, L.; He, Y.; Li, B.; Wang, L. Blueberry leaves as a promising sustainable source of polyphenols: Chemical composition, functional activities and future application perspectives. *Food Res. Int.* **2025**, *207*, 116110. [CrossRef]
14. Tobar-Bolaños, G.; Casas-Forero, N.; Orellana-Palma, P.; Petzold, G. Blueberry juice: Bioactive compounds, health impact, and concentration technologies—A review. *J. Food Sci.* **2021**, *86*, 5062–5077. [CrossRef]

15. Chen, W.; Liu, M.; Zhao, C.; Li, X.; Wang, Y. MTD-YOLO: Multi-task deep convolutional neural network for cherry tomato fruit bunch maturity detection. *Comput. Electron. Agric.* **2024**, *216*, 108533. [CrossRef]
16. Zhu, X.; Chen, F.; Zheng, Y.; Chen, C.; Peng, X. Detection of Camellia oleifera fruit maturity in orchards based on modified lightweight YOLO. *Comput. Electron. Agric.* **2024**, *226*, 109471. [CrossRef]
17. Li, Z.; Xu, R.; Li, C.; Munoz, P.; Takeda, F.; Leme, B. In-field blueberry fruit phenotyping with a MARS-PhenoBot and customized BerryNet. *Comput. Electron. Agric.* **2025**, *232*, 110057. [CrossRef]
18. Yang, W.; Ma, X.; An, H. Blueberry Ripeness Detection Model Based on Enhanced Detail Feature and Content-Aware Reassembly. *Agronomy* **2023**, *13*, 1613. [CrossRef]
19. Quiroz, I.A.; Alférez, G.H. Image recognition of Legacy blueberries in a Chilean smart farm through deep learning. *Comput. Electron. Agric.* **2020**, *168*, 105044. [CrossRef]
20. Diwan, T.; Anirudh, G.; Tembhurne, J.V. Object detection using YOLO: Challenges, architectural successors, datasets and applications. *Multimed. Tools Appl.* **2023**, *82*, 9243–9275. [CrossRef] [PubMed]
21. Ge, Z.; Liu, S.; Wang, F.; Li, Z.; Sun, J. Yolox: Exceeding yolo series in 2021. *arXiv* **2021**, arXiv:2107.08430. [CrossRef]
22. Kamalesh Kanna, S.; Ramalingam, K.; Pazhanivelan, P.; Jagadeeswaran, R.; Prabu, P.C. YOLO deep learning algorithm for object detection in agriculture: A review. *J. Agric. Eng.* **2024** , *55*, 1641. [CrossRef]
23. Terven, J.; Córdova-Esparza, D.M.; Romero-González, J.A. A comprehensive review of yolo architectures in computer vision: From yolov1 to yolov8 and yolo-nas. *Mach. Learn. Knowl. Extr.* **2023**, *5*, 1680–1716. [CrossRef]
24. Song, Z.; Li, W.; Tan, W.; Qin, T.; Chen, C.; Yang, J. LBSR-YOLO: Blueberry health monitoring algorithm for WSN scenario application. *Comput. Electron. Agric.* **2025**, *238*, 110803. [CrossRef]
25. Liu, S.; Qi, L.; Qin, H.; Shi, J.; Jia, J. Path aggregation network for instance segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 8759–8768.
26. Liu, C.; Wang, K.; Li, Q.; Zhao, F.; Zhao, K.; Ma, H. Powerful-IoU: More straightforward and faster bounding box regression loss with a nonmonotonic focusing mechanism. *Neural Netw.* **2024**, *170*, 276–284. [CrossRef]
27. Sun, Y.; Wang, J.; Wang, H.; Zhang, S.; You, Y.; Yu, Z.; Peng, Y. Fused-IoU loss: Efficient learning for accurate bounding box regression. *IEEE Access* **2024**, *12*, 37363–37377. [CrossRef]
28. Cheng, B.; Girshick, R.; Dollár, P.; Berg, A.C.; Kirillov, A. Boundary IoU: Improving object-centric image segmentation evaluation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Online, 19–25 June 2021; pp. 15334–15342.
29. He, K.; Gkioxari, G.; Dollár, P.; Girshick, R. Mask r-cnn. In Proceedings of the Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 2961–2969.
30. Zhang, H.; Wang, Y.; Dayoub, F.; Sunderhauf, N. Varifocalnet: An iou-aware dense object detector. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 19–25 June 2021; pp. 8514–8523.
31. Tian, Y.; Wang, S.; Li, E.; Yang, G.; Liang, Z.; Tan, M. MD-YOLO: Multi-scale Dense YOLO for small target pest detection. *Comput. Electron. Agric.* **2023**, *213*, 108233. [CrossRef]
32. Girshick, R. Fast r-cnn. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 1440–1448.
33. 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 Computer Vision—ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, 11–14 October 2016; Proceedings, Part I 14; Springer: Berlin/Heidelberg, Germany, 2016; pp. 21–37.
34. Ajayi, O.G.; Ashi, J.; Guda, B. Performance evaluation of YOLO v5 model for automatic crop and weed classification on UAV images. *Smart Agric. Technol.* **2023**, *5*, 100231. [CrossRef]
35. Shi, H.; Li, L.; Zhu, S.; Wu, J.; Hu, G. FeYOLO: Improved YOLOv7-tiny model using feature enhancement modules for the detection of individual silkworms in high-density and compact conditions. *Comput. Electron. Agric.* **2025**, *231*, 109966. [CrossRef]
36. Saha, S.; Noguchi, N. Smart vineyard row navigation: A machine vision approach leveraging YOLOv8. *Comput. Electron. Agric.* **2025**, *229*, 109839. [CrossRef]
37. Yi, W.; Xia, S.; Kuzmin, S.; Gerasimov, I.; Cheng, X. RTFVE-YOLOv9: Real-time fruit volume estimation model integrating YOLOv9 and binocular stereo vision. *Comput. Electron. Agric.* **2025**, *236*, 110401. [CrossRef]
38. Wang, A.; Chen, H.; Liu, L.; Chen, K.; Lin, Z.; Han, J.; Ding, G. Yolov10: Real-time end-to-end object detection. *Adv. Neural Inf. Process. Syst.* **2024**, *37*, 107984–108011.
39. Khanam, R.; Hussain, M. Yolov11: An overview of the key architectural enhancements. *arXiv* **2024**, arXiv:2410.17725. [CrossRef]
40. Zhang, Q.; Rao, L.; Yang, Y. Group-cam: Group score-weighted visual explanations for deep convolutional networks. *arXiv* **2021**, arXiv:2103.13859.

41. Selvaraju, R.R.; Cogswell, M.; Das, A.; Vedantam, R.; Parikh, D.; Batra, D. Grad-cam: Visual explanations from deep networks via gradient-based localization. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 618–626.
42. Wang, C.; Han, Q.; Li, J.; Li, C.; Zou, X. YOLO-BLBE: A Novel Model for Identifying Blueberry Fruits with Different Maturities Using the I-MSRCR Method. *Agronomy* **2024**, *14*, 658. [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.