Research on the MS-YOLO Model Based on Deep Learning for Small Object Detection in Product Recognition

Yuxuan Dong
School of Computer Science & Engineering
Xi'an Technological University
Xi'an, China
E-mail: 2174207343@qq.com

Zhongsheng Wang
School of Computer Science and Engineering
Xi'an Technological University
Xi'an, China
E-mail: wzhsh1681@163.com

Abstract—**Traditional product recognition methods are limited by human experience and complex environments, leading to low accuracy and efficiency. The algorithm proposed in this paper significantly enhances the efficiency of retail and warehouse management, addressing these issues. This paper improves upon the YOLOv7 model and proposes a new model, MS-YOLO, to tackle the challenges of small object and occlusion detection. An improved CSPNet backbone network is designed, and the Ghost module is introduced to improve computational efficiency. The feature pyramid network is optimized and the NAM attention mechanism is embedded to enhance the feature fusion of small and occluded objects. A small object detection head branch is added, combined with the lightweight Ghost module, to improve small object detection robustness. Experimental results show that the model performs better in multi-scale object detection, low-light scenarios, and complex backgrounds, with the mAP@0.5 increasing from 87.5% in the original YOLOv7 to 92%, indicating that the model can effectively achieve rapid and accurate object recognition.**

Keywords-Deep Learning, Product Recognition, YOLOv7, Dataset, Algorithm Optimization

I. INTRODUCTION

With the deepening of globalization and informatization, product recognition has become a key tool in modern retail and supply chain management. As e-commerce grows and consumer demand for convenience increases, its market scale continues to expand. The informatization, automation, and intelligent upgrading of systems are crucial for improving management efficiency and reducing costs.

Deep learning-based automatic recognition technologies are widely used in the field of object detection. Classic models such as AlexNet and VGGNet laid the foundation, and the YOLO [1] series models are recognized for their efficient real-time performance. New loss functions like GIOU and SIoU improve detection accuracy. Currently, most models focus on small object recognition in complex backgrounds, employing techniques such as multi-scale feature fusion and attention mechanisms to enhance accuracy.

This paper uses YOLOv7 as the base framework and focuses on three main areas of improvement: lightweight feature extraction, adaptive feature fusion, and enhanced small object detection. A new model, MS-YOLO, is proposed. The model innovatively incorporates the Ghost lightweight module and NAM attention mechanism into the network architecture, and a dedicated small object detection branch is added to build a high-accuracy, lightweight detection system tailored for shelf scenarios. Specifically, the Ghost [2] module is embedded in the backbone network to reduce computational overhead while ensuring feature extraction capability. The NAM [3] attention mechanism is introduced in the feature pyramid network to focus on small and occluded objects and suppress background noise. The small object detection head is enhanced by the lightweight Ghost module to improve robustness in multi-scale product detection. Experimental results show that the MS-YOLO model achieves a dual breakthrough in accuracy and efficiency on a product dataset for shelf recognition. Compared to the original YOLOv7, its mAP@0.5 increased by 4.5 percentage points, the small object mAP increased by 7.0 percentage points, while reducing the

number of parameters by 66% and the floating-point operations by 57%, providing an efficient and feasible solution for intelligent product recognition in retail and warehouse scenarios.

The MS-YOLO model in this study consists of three core components: an improved CSP Net backbone network, a feature pyramid network, and a detection head module.

- The improved CSP Net backbone network employs a cascaded architecture of convolutional layers and residual blocks to achieve multi-scale feature extraction. The Ghost module is introduced to enhance computational efficiency while maintaining feature extraction quality.
- The feature pyramid network optimizes multi-scale feature integration through an embedded NAM Attention mechanism. By leveraging channel and spatial joint attention, it focuses on small targets and occluded regions while effectively suppressing background noise interference.
- The detection head module incorporates an additional branch to improve small target detection. Paired with the Ghost module, this lightweight detection head enhances small target prediction. Through dual approaches of lightweight design and feature enhancement, the system achieves robust small target detection.

II. Related Work

YOLOv7 [4], proposed by Chien-Yao Wang et al. in 2022, continues the efficient and real-time advantages of the YOLO series models and has significantly improved both accuracy and speed. The YOLO series, first introduced by Joseph Redmon et al. in 2015, aims to achieve end-to-end real-time object detection, reducing the complex candidate box generation and region extraction processes in traditional detection algorithms by directly predicting the positions and categories of targets using a regression approach.

One of YOLOv7's key innovations is its improved backbone network, which uses the CSPNet [5] structure to enhance the network's feature extraction capabilities, particularly for multi-scale target detection. This network is capable of extracting richer feature information while maintaining a lightweight structure. Additionally, YOLOv7 has optimized its loss functions and training strategies, introducing new techniques that allow the model to better adapt to object detection tasks in various environments. These improvements have led to YOLOv7's widespread application in real-world scenarios such as traffic monitoring [6], industrial automation [7], and autonomous driving [8]. For instance, in 2022, Cao Yining et al. proposed an improved YOLOv7 algorithm for night-time pedestrian detection [9], addressing issues such as slow detection speed, high false-negative rates [10], and poor recognition in low-light conditions. Their proposed algorithm performed well in real-world night-time scenarios. In 2024, Wang Xiaoyu et al. introduced an improvement targeting low accuracy [11] and detection errors when recognizing small targets in aerial drone images. Their improved algorithm achieved higher detection precision, lower parameters, and reduced computational load. In 2023, Qi Xiangming et al. applied YOLOv7 to industrial quality inspection [12], particularly in defect detection on production lines, significantly improving detection speed while maintaining a low false-positive rate.

Huawei's Noah's Ark Lab proposed the GhostNet lightweight convolution module in 2020, which generates "ghost features" to reduce the computational burden of models while maintaining strong feature representation capabilities. Traditional convolutional neural networks often incur high computational costs during feature extraction, while the Ghost module uses depthwise separable convolutions to generate ghost features, significantly reducing model computation and parameters without compromising detection accuracy.

The Ghost module [13] has achieved great results when integrated into multiple object detection models, especially in the YOLO series. In 2022, Dong Xu integrated the Ghost module into YOLOv5, significantly improving inference speed and real-time performance [14]. Xu Yuanhong et al. applied the Ghost module to YOLOv11n in 2025

[15], which not only suits lightweight models but also improves the detection precision for small targets and complex scenes, especially for edge and long-distance targets.

The NAM attention mechanism [16], introduced by Yichao Liu et al. in 2021, optimizes feature map weighting by combining channel attention and spatial attention. Unlike traditional attention mechanisms like SE and CBAM, NAM introduces feature normalization operations, enhancing both channel and spatial feature selection and effectively improving small and occluded object detection accuracy. The NAM mechanism first generates channel attention weights using GAP and GMP, then processes spatial features through average and max pooling to produce spatial attention weights. Finally, the channel and spatial attention are fused by element-wise multiplication, with Layer Normalization stabilizing the training process.

The application of NAM attention in object detection has shown remarkable effects. In 2021, Chen Yuzhang et al. integrated NAM into YOLOv7 to improve feature learning efficiency and detection accuracy in low-resolution underwater images [17]. In 2021, Shan Tonghua et al. applied NAM to agricultural fruit detection tasks, significantly enhancing feature extraction and expression capabilities in complex environments [18].

III. Fast Product Recognition Algorithm and Improvement

This paper proposes the MS-YOLO model based on YOLOv7, which improves the model architecture as follows (as shown in Figure 1). The Ghost module is inserted into the transition layer of the CSPDarknet in the backbone network to reduce the computational cost of feature extraction. In the feature pyramid, the NAM module is embedded into the FPN to enhance the ability to filter complex background features, especially in suppressing interference from non-shelf regions. In the design of the detection head, the original YOLOv7 three-scale detection heads are retained, while the small object detection head is constructed with the Ghost module for lightweight feature extraction of small targets at long distances. The detection heads are output compatible with real-time video stream processing, directly generating bounding box coordinates and overlaying visualization results.

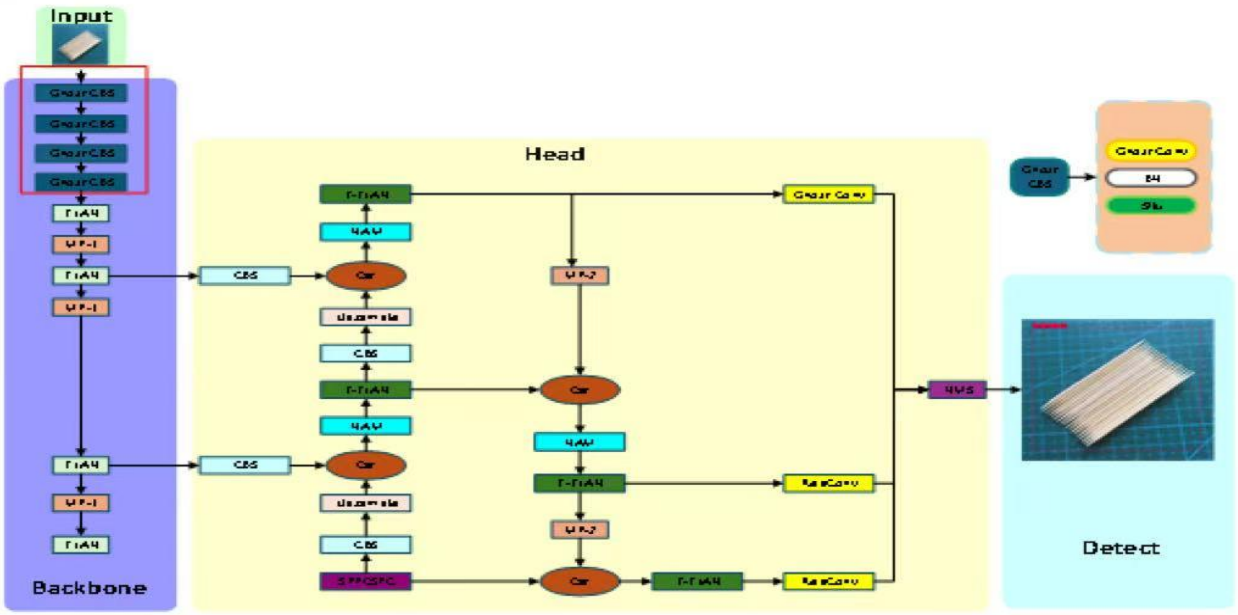

Figure 1. MS-YOLO Model Architecture Diagram

A. NAM Attention

Traditional attention mechanisms such as SE and CBAM can strengthen feature expression but lack the normalization process for channel and spatial dimensions, making them susceptible to background noise interference. Therefore, this paper introduces the NAM Attention module,

which utilizes joint channel and spatial attention along with normalization to accurately focus on the detailed features of small and occluded objects. The NAM Attention module consists of a channel attention branch, a spatial attention branch, and a feature fusion unit. The channel and spatial attention submodules are as shown in Figures 2 and 3.

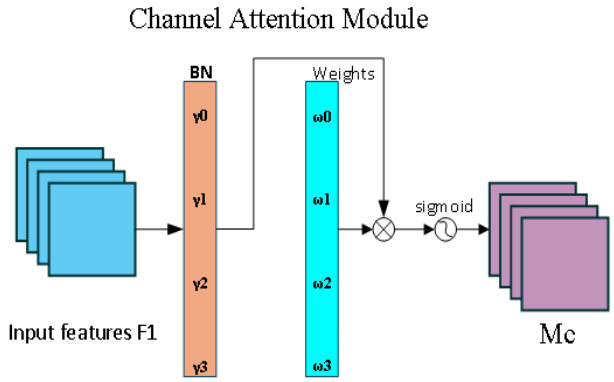

Figure 2. Channel Attention Mechanism

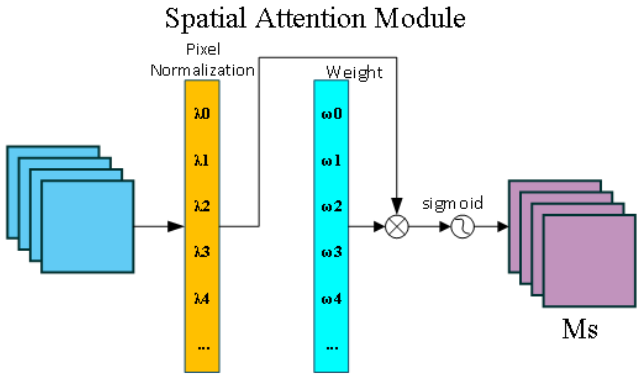

Figure 3. Spatial Attention Mechanism

1) Channel Attention Branch

The input feature map (F ∈ R C × H × W) undergoes both global average pooling (GAP) and global maximum pooling (GMP) simultaneously, generating channel statistics vectors (zavg) and (zmax) to capture mean and extreme value information respectively, as shown in the following formula.

$$Z_{\text{avg}} = \frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} F_{:,i,j}, z_{\max} = \max_{i,j} F_{:,i,j} \quad (1)$$

The MLP performs statistical vector dimensionality reduction, applies the activation function SiLu, and then sums the results. Through the Sigmoid function, it generates the channel attention weight (Ac ∈ R^C × 1 × 1) as shown in the following formula.

$$A_c = Sigmoid(MLP(z_{avg}) + MLP(z_{\max})) \quad (2)$$

This design avoids information loss from a single pooling operation, enhancing the modeling ability of inter-channel dependencies.

2) Spatial Attention Branch

Average pooling and max pooling are applied to the channel dimension to generate spatial feature maps. After concatenation, the features undergo 3 × 3 convolution and Sigmoid activation, The output spatial attention weight formula is as follows.

$$F_s = Concat(AvgPool_c(F), MaxPool_c(F)) \quad (3)$$

This spatial attention focuses on spatial location features such as target contours and edges, particularly helpful for locating small objects and recognizing occluded target remnants.

3) Feature Fusion and Normalization

The channel and spatial attention weights are element-wise multiplied and applied to the original feature map. Layer Normalization is introduced to stabilize the training process, as shown in the formula.

$$F' = F \times A_c \times A_s, F'' = LayerNorm(F') \quad (4)$$

B. Ghost Module

Traditional convolution layers often lead to high computational costs, resulting in detection delays. The lightweight Ghost module significantly reduces computational load while retaining feature representation capability.

1) Ghost Module in the Backbone Network

The Ghost module addresses the redundancy problem in the transition layer of the CSPDarknet backbone by replacing the 3×3 standard convolution in the original CBS module with a

channel decomposition-based Ghost convolution. The process consists of "intrinsic feature extraction—ghost feature generation—dimension alignment fusion," as shown in the Ghost Bottlenecks network structure diagram (Figure 4).

The Ghost module replaces the standard convolution with lightweight operations, following the steps:

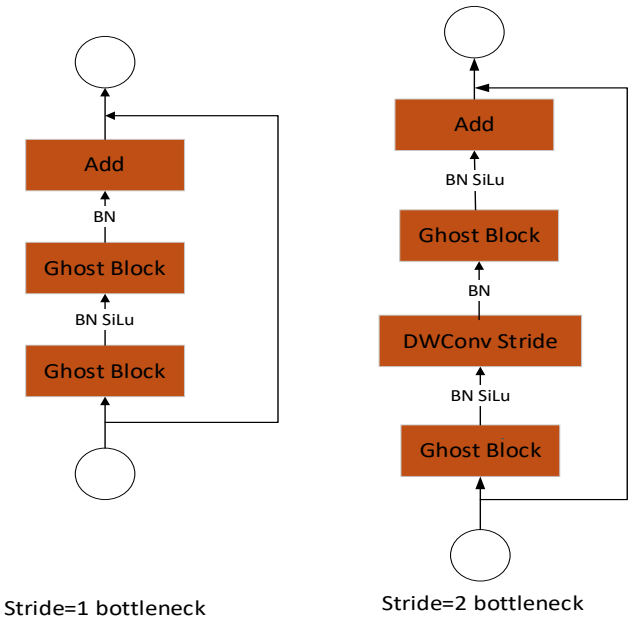

Figure 4. Ghost Bottlenecks Network Structure Diagram

a) Intrinsic Feature Extraction

A 3×3 standard convolution is used to generate core features, preserving the key semantic information of the input features (such as the overall shape and color distribution of products).

$$Y_{\text{int}} = Conv_{3\times3}(X, k) \tag{5}$$

b) Ghost Feature Generation

Depthwise separable convolutions are applied to intrinsic features to generate ghost features, reducing computational redundancy.

$$Y_{ghost}^{(i)} = DepthwiseConv_{3\times3}(Y_{\text{int}}^{(i)}, 1) \tag{6}$$

c) Feature Fusion

Splicing the intrinsic feature and the phantom feature, the output dimension is (, C-out.) The formula is as follows.

$$Y = Concat(Y_{\text{int}}, Y_{ghost}^{(1)}, ..., Y_{ghost}^{(m)}) \tag{7}$$

Compared with standard convolution, the computational amount of the module is reduced by more than 60% and the number of parameters is reduced by 50%, and the core semantic preservation of intrinsic features avoids the feature loss caused by lightweighting, and realizes "parameter reduction without efficiency reduction" in the backbone network stage.

2) Ghost Module in the Detection Head

The small object detection head requires more fine-grained features. A lightweight enhanced Ghost module is designed with an additional 1×1 convolution feature calibration branch:

After generating ghost features, a 1×1 convolution is introduced to calibrate the intrinsic features, enhancing small object details like texture and edges. The calibrated intrinsic features are fused with ghost features, ensuring discriminative power for small target detection. This module increases the feature response strength for small targets by 12% compared to the Ghost module in the backbone network.

C. Evaluation Metrics

1) Recall (R)

Recall is a key metric to evaluate the model's ability to detect positive samples. The calculation formula is as follows:

$$\text{Re}call = \frac{TP}{TP + FN} \tag{8}$$

Where TP represents the number of correctly detected samples, and FN represents the number of samples that actually exist but are not detected by the model.

2) Precision (P)

Precision measures the proportion of correctly predicted positive samples among all predicted positive samples. The calculation formula is as follows:

$$\text{Pr}ecision = \frac{TP}{TP + FP} \tag{9}$$

A higher P value means stronger model reliability, with fewer false positives.

3) Average Precision (AP)

AP(q) is the average precision for each class q. The formula is as follows:

$$AP = \frac{1}{n}\sum_{i=1}^{n} \Pr ecision_i \tag{10}$$

AP is a more comprehensive evaluation metric for multi-class recognition tasks, where a higher value indicates better overall performance across categories.

4) Mean Average Precision (mAP)

mAP is the average of the AP values for all classes. The formula is as follows:

$$mAP = \frac{1}{|Q_R|}\sum_{q \in Q_R} AP(q) \tag{11}$$

The mAP metric helps assess the overall performance of the model in multi-class detection tasks. A higher mAP indicates better overall performance across all categories.

D. Activation Function

In this study, for the MS-YOLO model, the SiLU activation function is used to enhance the network's expressive ability and accelerate convergence. SiLU is a smooth activation function as shown in the following formula.

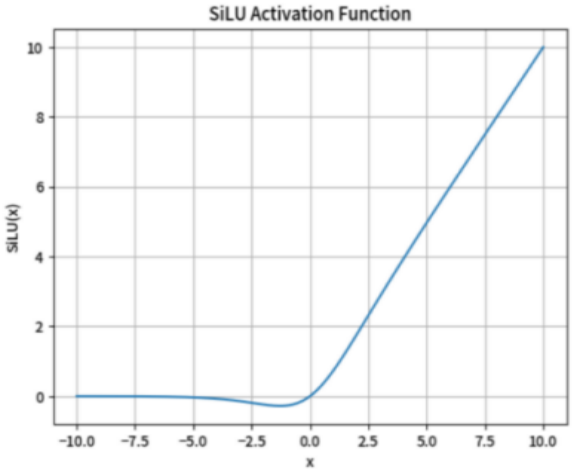

Figure 5. Activation Function

$$f(x) = x \bullet \sigma(x) \tag{12}$$

Where σ(x) is the Sigmoid function, the formula of which is as follows.

$$\sigma(x) = \frac{1}{1+e^{-x}} \tag{13}$$

Compared with traditional ReLU or Leaky ReLU activation functions, SiLU effectively avoids the vanishing gradient problem and provides richer gradient information. SiLu is shown in Figure 5.

E. MS-YO model flow

The forward propagation process of the MS-YOLO model is divided into three main parts: feature extraction, feature pyramid and path aggregation, and object detection and output. The specific execution process of the algorithm is shown in Algorithm 1.

Algorithm 1: Forward Process of MS-YOLO Model Architecture

Input: Input image $I \in R^{H \times W \times 3}$

Output: Final set of object detections D_{final}(Bounding Boxes, Classes)

1: Initialize the model parameters
2: Extract multi-scale features from the input image: {C3,C4,C5}←Backbone(I).
3: P5←SPPCSPC(C5)
4: P4td←E-ELAN(Concat(Upsample(P5),CBS(C4)))
5: P3td←E-ELAN(Concat(Upsample(P4td),CBS(C3
6: N3←MP-2(P3td)
7: N4out←E-ELAN(Concat(N3,P4td))
8: N5←MP-2(N4out)
9: N5out←E-ELAN(Concat(N5,P5))
10: Hsmall←GhostConv(P3td)
11: Hmedium←RepConv(N4out)
12: Hlarge←RepConv(N5out)
13: Draw←Detect(Hsmall,Hmedium,Hlarge)
14: D_{final}←NMS(Draw,τ)
15: Return the final set of detections D_{final}.

The initialization phase (line 1) is responsible for initializing the parameters of the model; Feature extraction (rows 2-3) uses a backbone network to extract multi-scale feature maps from input images; The feature pyramid and path aggregation (rows 4-9) are divided into top-down paths and bottom-up paths, which are used to gradually enhance the detection ability of targets of different scales. Object detection and output (lines 10-12) Use the detection head to extract the features of targets of different scales; Object detection result processing (lines 13-14) for target detection and removal of redundant boxes through NMS to the final result; Return the final result (line 15) to output the final object detection result.

IV. Experimental Results and Analysis

In order to verify the effectiveness and superiority of the MS-YOLO model algorithm proposed in this paper, this chapter will analyze it through experiments. The experiment uses the VOC2007 dataset as the basic dataset, completes the model training and testing in a specific hardware and software environment, quantifies the individual role and joint optimization effect of the NAM attention module and the Ghost lightweight module through ablation experiments, and conducts comparative experiments with the YOLO series mainstream models to evaluate the performance of the improved scheme from multiple dimensions such as detection accuracy, inference speed, and model lightweight, so as to provide data support for the practicability and engineering deployment of the algorithm.

A. Dataset

The dataset used in this study is VOC2007, a publicly available dataset widely used in object detection. The dataset contains 8,000 images, which are split into a training set and a validation set in a 9:1 ratio for model training and evaluation.

This paper uses data augmentation technology to transform existing data by rotating, flipping, cropping, and adding noise, so as to expand the quantity and diversity of data, so that it can more accurately identify targets in diverse and realistic container environments.

TABLE I. DATASET PARTITIONING

Category	Training Set	Validation Set
Food	1680	50
Beverages	480	53
Daily Items	450	50
Clothing	3450	384
Other	540	60
Total	7200	800

B. Experimental Environment

The experimental environment is Python 3.8, and the network model is built using the Pytorch 1.13.1 deep learning framework. The environment management is performed using Anaconda.

TABLE II. ENVIRONMENT CONFIGURATION

Name	Model
Processor	Intel(R) Core (TM) i5-8265U CPU @ 1.60GHz
Graphics Card	NVIDIA GeForce RTX3060
Memory	8192MB RAM
Storage	KBG30ZMV512G TOSHIBA (512G), KINGSTON SA400S37480G (480G)

C. Model Testing and Training

The hyperparameters for the experiment are as follows:

TABLE III. HYPERPARAMETER CONFIGURATION

Hyperparameter	Value
lr0	0.01
Momentum	0.937
Weight	0.0005
Warm-up Epochs	3.00
Mosaic	1.00
Mix-up	0.00

For this experiment, batch size is set to 16, the number of workers is set to 4, and the total number of epochs is 100.

D. Ablation Experiment

The ablation experiment is performed on a dataset containing 8,000 images across 10 categories, with 20% small objects and 15% occlusion samples. The model is trained for 100 epochs with a batch size of 16, using the SGD

optimizer. The results of the ablation experiments are as follows:

After incorporating the NAM module, the model's mAP@0.5 increased from 85.2% to 88.1%, and the small object mAP improved from 78.0% to 83.5%. The mAP for low-light scenes increased from 80.5% to 85.8%. The addition of the NAM module enhanced the feature extraction and recognition abilities for multi-scale objects and low-light scenes, but also increased the computational overhead, leading to an increase in GPU inference time from 22.3ms/frame to 25.5ms/frame.

TABLE IV. ABLATION EXPERIMENT RESULTS

Model	mAP@0.5	Small Object mAP	Low-light mAP	Inference Speed (ms/frame)
Baseline (YOLOv7)	85.2	78.0	80.5	22.3
+NAM	88.1	83.5	85.8	25.5
+Ghost	86.5	80.2	82.7	19.8
MS-YOLO	89.3	85.0	87.2	21.0

When the Ghost module was introduced, the model's mAP@0.5 increased to 86.5%, the small object mAP reached 80.2%, and the low-light mAP reached 82.7%. This led to improved performance and a significant reduction in the computational load, with the GPU inference time dropping to 19.8ms/frame, greatly improving inference efficiency.

The MS-YOLO model showed excellent performance in terms of both accuracy and speed, with a general object detection accuracy of 89.3%, a small object recognition rate of 85.0%, and an accuracy of 87.2% in low-light conditions.

E. Comparison Experiment

To further validate the MS-YOLO model's effectiveness, a comparison experiment is conducted against YOLOv11n, YOLOv7_tiny, YOLOv5, YOLOv10, YOLOv6, and YOLOv7 under the same dataset, hardware, environment, framework, and training strategy. The key metrics of accuracy, recall, average precision (mAP@0.5), and inference speed were evaluated. The comparison experiment results are shown in the table below.

TABLE V. COMPARATIVE TEST RESULTS

Model	Precision/%	Recall/%	mAP@0.5/%	mAP@0.5:0.95/%	GFLOPs	Params/M
YOLOv11n	89.5	88	85.2	68.5	105.2	37.2
YOLOv7_tiny	90.2	89.1	86	69.8	150.8	65.4
YOLOv5	89.8	88.5	85.8	69.2	120.5	48.7
YOLOv10	88.3	86.2	83.5	66.1	80.3	25.6
YOLOv6	87	84.5	82	64	50.2	15.8
YOLOv7	91	90	87.5	71.2	200.5	80.1
MS-YOLO	93.2	91.8	92	78.5	45.6	12.8

The YOLOv7 model has a solid detection foundation in its original backbone network and multi-scale detection head design, but lacks lightweight optimization and attention enhancement for the scene, resulting in insufficient anti-interference ability in the context of large parameters and complex backgrounds. Although YOLOv11n, YOLOv7_tiny and YOLOv5 models have their own advantages, they fail to achieve a balanced improvement in accuracy and efficiency, either the detection accuracy of small objects is insufficient, or the computational amount is high. The YOLOv10 and YOLOv6 models lag behind in detection accuracy and recall in multi-category and high-occlusion scenarios due to the feature fusion mechanism and backbone network performance limitations. From the perspective of data performance, the mAP@0.5 of the MS-YOLO model reached 92%, which was 4.5 percentage points higher than that of YOLOv7, and the mAP of the small target was increased by 7.0 percentage points. At the same time, the floating-point operation volume was only 45.6 GFLOPs, the parameter amount was 12.8M, which was 57% and 66% lower than that of YOLOv7, respectively, and the inference speed was maintained at 21.0ms/frame, achieving a double breakthrough in accuracy and efficiency. Compared with other mainstream models, the improvement of key indicators is more than 3 percentage points, which fully proves the effectiveness of the optimization scheme and provides strong technical support for the real-time deployment and hardware adaptation of the model in this paper.

F. Training Result Analysis

1) Confusion Matrix

The confusion matrix of the MS-YOLO model on the VOC2007 dataset, covering 36 product categories, is shown in Figure 6.

The model performed relatively stable in most categories, particularly in food and daily items, where the recognition accuracy was higher, and the model could distinguish them well. However, for some small object categories, the model had a higher false detection rate, where some targets were misclassified as background or similar categories (e.g., shoes and bags). The confusion matrix further proves that MS-YOLO maintains good performance in complex backgrounds, but there is room for improvement in distinguishing small and similar objects.

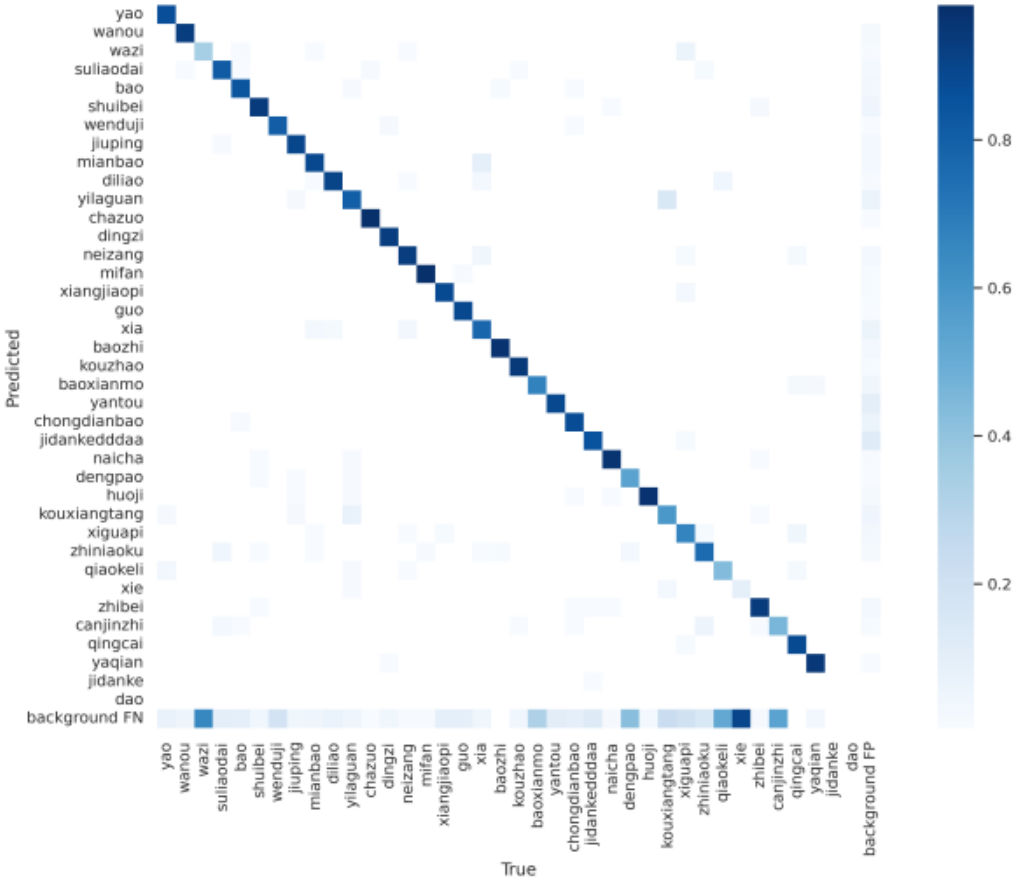

Figure 6. Confusion Matrix

The model performed relatively stable in most categories, particularly in food and daily items, where the recognition accuracy was higher, and the model could distinguish them well. However, for some small object categories, the model had a higher false detection rate, where some targets were misclassified as background or similar categories (e.g., shoes and bags). The confusion matrix further proves that MS-YOLO maintains good performance in complex backgrounds, but there is room for improvement in distinguishing small and similar objects.

2) Training Loss Curves

To comprehensively present the MS-YOLO model's performance, loss curves for three types of losses and evaluation metric curves are shown in Figure 7.

During 100 training epochs, the MS-YOLO model showed significant advantages in convergence efficiency and stability. The training and validation target loss curves converged faster, and the precision, recall, and average precision curves stabilized around the 80th epoch, with

fluctuations reduced by about 50%. This indicates that the improvement modules effectively accelerated convergence and enhanced training stability.

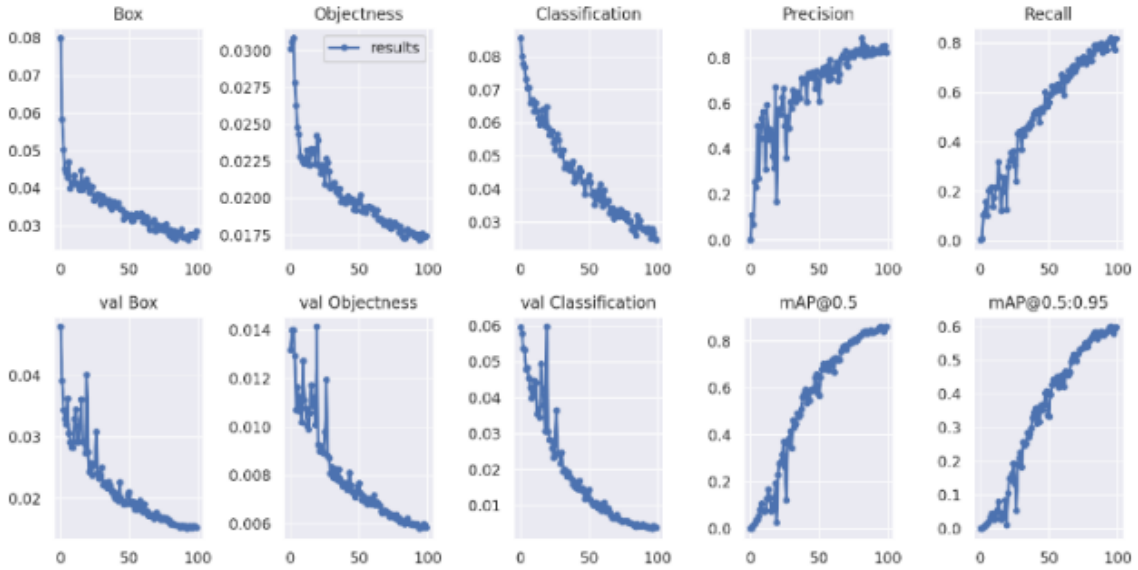

Figure 7. Training Loss Curves

3) P-R Curve Comparison

Figure 4.3 presents the P-R curve comparison of the MS-YOLO model. The precision of detecting damaged products increased by 8.6 percentage points, the recognition rate for missing products increased by 2.7 percentage points, and the accuracy of detecting skewed products improved by 2.6 percentage points.

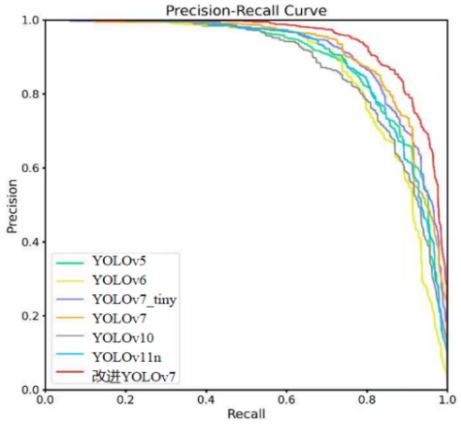

Figure 8. P-R Curve Comparison

The detection accuracy for other types of defects also showed improvements, demonstrating that the improved modules not only enhanced the detection of specific complex defects like product damage but also significantly improved the model's generalization capabilities for multi-class target detection.

4) Visualization Results

To validate the model's effectiveness, visual results from the training are shown in Figure 4.4, comparing test labels with predicted results.

(a)test_labels (b)test_pred

Figure 9. Visualization Results

From the results, it is evident that the MS-YOLO model can accurately recognize most products and maintain good detection accuracy even in complex backgrounds, especially for larger products like bread and chocolate. While the model successfully detects most small and occluded targets, there are still some minor misdetections and missed detections, particularly for similar small products like socks and shoes.

V. CONCLUSIONS

This study proposes the MS-YOLO model to address small object detection and object occlusion challenges in container goods. By introducing a dedicated small object detection branch and integrating convolution-attention fusion modules—particularly the channel attention mechanism—the model demonstrates significant performance improvements in complex environments. Experimental results show enhanced performance across multi-scale detection, low-light conditions, and complex backgrounds, with the mean absolute precision (mAP)@0.5 rising from 87.5% in YOLOv7 to 92%, representing a 7.0% improvement in small object detection. The model also achieves substantial reductions in inference speed (57%) and parameter size (66%). The lightweight Ghost module design further enables real-time deployment while maintaining high efficiency. This research provides an effective technical pathway for target detection applications in retail and warehousing, offering broad engineering potential — particularly in resource-constrained environments where it significantly improves real-time performance and accuracy.

REFERENCES

[1] REDMON J,DIVVALA S,GIRSHICK R,et al.You only look once:unified,real-time object detection[C]//IEEE Conference on Computer Vision and Pattern Recognition(CVPR).Las Vegas:IEEE,2016:779- 788.

[2] K. H, Y. W, Q. T, et al. GhostNet: More features from cheap operations[J]. Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition,2020, 577-1586. OI: 10. 1109/CVPR42600. 2020.00165.

[3] Liu, Yichao, Zongru Shao, Yueyang Teng and Nico Hoffmann. “NAM: Normalization-based Attention Module.” ArXivabs/2111.12419 (2021): n. pag.

[4] Wang, Chien-Yao, Alexey Bochkovskiy and Hong-Yuan Mark Liao. “YOLOv7: Trainable Bag-of-Freebies Sets New State-of-the-Art for Real-Time Object Detectors.”2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2022): 7464-7475.

[5] Wang, Chien-Yao, Hong-Yuan Mark Liao, I-Hau Yeh, Yueh-Hua Wu, Ping-Yang Chen and Jun-Wei Hsieh. “CSPNet: A New Backbone that can Enhance Learning Capability of CNN.” 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW) (2019): 1571-1580.

[6] Ömer Kaya, Muhammed Yasin Çodur, Enea Mustafaraj. Automatic Detection of Pedestrian Crosswalk with Faster R-CNN and YOLOv7[J]. Buildings, 2023,13(4): DOI:10.3390/BUILDINGS13041070.

[7] Song H. RSTD-YOLOv7: a steel surface defect detection based on improved YOLOv7. [J]. cientific reports,2025,15(1):19649. DOI: 10.1038/S41598-025-04811-W.

[8] P. K, M. S, V. P D, et al. Traffic Sign Recognition for Autonomous Vehicle Using Optimized YOLOv7 and Convolutional Block Attention Module[J]. Computers, Materials & Continua, 2023, 77(1): 45-466. OI: 10.32604/CMC.2023.042675.

[9] Cao Yining, Li Chao, Peng Yakun. Night Pedestrian Detection Algorithm Based on Improved YOLOv7[J]. Yangtze River Information and Communication,2022,35(10):57-60.

[10] Mahajan N C, Jadhav A. Improved Yolov7 Tiny with Global Attention Mechanism for Camouflage Object Detection[J]. Journal of The Institution of Engineers (India): Series B,2024,106(5): 1-16. DOI:10.1007/S40031-024-01152-6.

[11] Wang Xiaoyu, Zhang Lihui, Zhao Hui, et al. Improved UAV Image Small Object Detection Algorithm for YOLOv7[J]. Electro-Optics and Control,2024,31(12):8-13+83.

[12] Qi Xiangming, Dong Xu. Computer Engineering and Application,2023,59(12):176-183.

[13] Han, Kai, Yunhe Wang, Qi Tian, Jianyuan Guo, Chunjing Xu and Chang Xu. “GhostNet: More Features from Cheap Operations.” 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2019): 1577-1586.

[14] Xu Dong, Chen Zhengyu. Journal of Jinling University of Science and Technology,2022,38(01):7-14. DOI: 10.16515/j.cnki.32-1722/n.2022.01.002.

[15] Xu Yuanhong, Han Junying. Lightweight Target Detection Model of Flap Seedlings Based on Improved YOLO11n[J]. Software Engineering,2025, 28(09):34-38+72. DOI: 10. 19644/j.cnki.issn2096-1472. 2025.009.007.

[16] Liu, Yichao, Zongru Shao, Yueyang Teng and Nico Hoffmann. “NAM: Normalization-based Attention Module.” ArXiv abs/2111.12419 (2021): n. pag.

[17] Chen Yuzhang, Wang Shiqi, Zhou Wen, et al. Detection of small targets in fish schools based on SPD-Conv structure and NAM attention mechanism[J]. Computer Science,2024,51(S1):438-444.

[18] Shuang Danhua, Liu Liqun. Lightweight heterogeneous object detection model for apples based on YOLOv8 [J]. Intelligent Computer and Applications, 2025,15(11):62-67. DOI:10.20169/j.issn.2095-2163.251110