FG-YOLO: An Efficient Framework for Flexible Glove Surface Defect Detection

Yuhang Lu[1], Shenao Li[2], Qiuyan Wang[2], Yaheng Ren[3], Yongjiang Xue[2], Qingzeng Song[2*]
[1]School of Software, Tiangong University, Tianjin, China
[2]School of Computer Science and Technology, Tiangong University, Tianjin, China
[3]Institute of Applied Mathematics, Hebei Academy of Sciences, Shijiazhuang, China
Email: {2331111286, xueyongjiang}@tiangong.edu.cn, 1343081636@qq.com, yahengren@126.com,
{wangyan198801, qingzengsong}@163.com

***Abstract*—Flexible gloves are widely used in industrial production. However, their soft and deformable surfaces make subtle defects difficult to detect under complex backgrounds. To address this challenge, we propose FG-YOLO, an efficient and robust detection framework for real-time industrial glove surface defect detection. The proposed method redesigns the Spatial Fast Connection (SFC) module and the Bi-level Routing Attention (BRA) module, and proposes a novel Multi-Scale Cross Attention Network (MSCANet) to enhance multi-scale feature representation and suppress background interference. This design enables accurate perception of small and fine-grained defects while maintaining real-time performance. In addition, a high-quality industrial glove defect dataset is constructed using real production data to support reliable training and evaluation. Furthermore, a production-oriented automatic inspection system integrating multi-camera acquisition, real-time detection, and automated rejection is developed to validate the practicality of the proposed framework in continuous industrial environments. Experimental results show that FG-YOLO achieves 89.54% mAP50 and 60.59% mAP50:95 while maintaining an inference speed of 280 FPS, outperforming the baseline YOLO11n and demonstrating strong generalization across multiple datasets. These results indicate that the proposed method provides an effective and deployable solution for real-time industrial surface defect detection.**

***Index Terms*—Small Object Detection, Surface Defect Detection, Multi-Scale Feature Learning, Attention Mechanism, Industrial Visual Inspection, Real-Time Detection**

I. Introduction

Flexible gloves are widely used in medical, food processing, and electronic assembly industries due to their elasticity and close fit [1]. However, their soft and deformable surfaces are prone to subtle defects such as cracks, stains, and color variations, which are difficult to detect manually and may significantly affect protective performance. As industrial quality requirements continue to increase, accurate and efficient defect detection has become an essential component of automated inspection systems [2].

With the rapid development of deep learning, convolutional neural network–based detectors have been extensively applied to industrial visual inspection tasks. One-stage detectors represented by the YOLO family [3] achieve a favorable balance between accuracy and inference speed, making them suitable for real-time industrial deployment. Recent studies further improve efficiency and robustness through lightweight designs and end-to-end optimization. For example, YOLOv10 [4] reduces post-processing overhead, while lightweight variants of YOLOv8 [5] demonstrate strong performance in real-time small object detection scenarios.

Transformer-based detection frameworks have also attracted increasing attention due to their strong global modeling capability. RT-DETR [6] introduces efficient query mechanisms for real-time detection, while Swin Transformer [7] enhances multi-scale representation through shifted-window attention. Deformable attention [8] further improves localization accuracy with reduced computational cost. However, transformer-based models typically require substantial computation and memory resources, which limits their applicability in real-time industrial inspection scenarios.

In practical industrial surface detection, defect targets are often small and easily confused with background textures, especially for flexible materials with complex deformations. Although attention-enhanced feature fusion and multi-scale aggregation strategies have been widely studied to improve detection robustness [9]– [11], existing lightweight detectors still struggle to preserve fine-grained details under complex backgrounds, while heavier models fail to meet strict real-time constraints [12]. Recent surveys further highlight the importance of effective feature enhancement and cross-scene generalization for small object detection in industrial environments [13]– [18].

Moreover, high-quality datasets for flexible glove detection remain scarce, and defect instances are often limited in number and diversity, further increasing the difficulty of reliable model training and evaluation. Therefore, achieving a balanced trade-off among detection accuracy, computational efficiency, and generalization remains a key challenge in practical flexible material surface defect detection. To address these challenges, this work develops an efficient and robust detection framework that enhances multi-scale feature representation and background suppression while preserving real-time deployment capability. Based on this objective, this paper makes three main contributions:

- We propose FG-YOLO, a lightweight yet robust detection framework that enhances small-defect perception through redesigned SFC, newly developed MSCANet,

* Corresponding author.

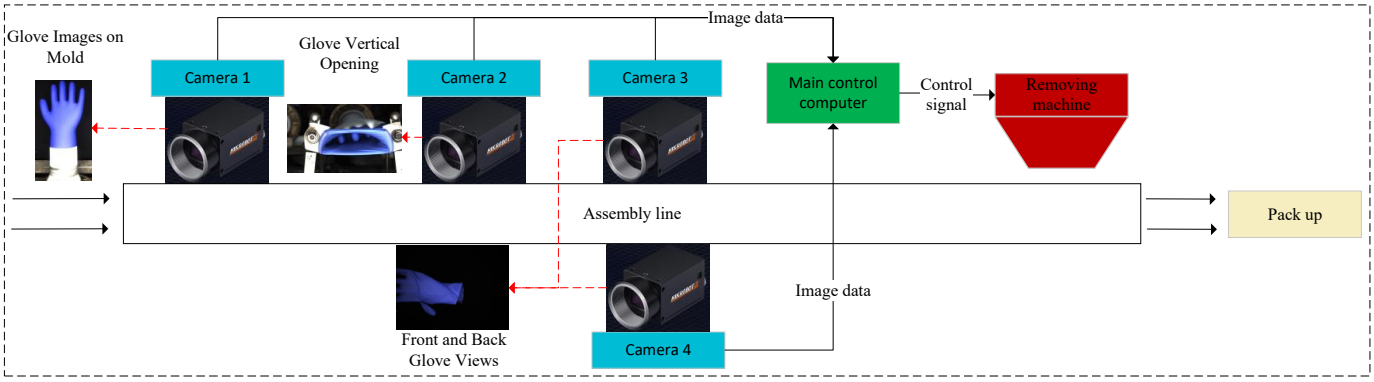


Fig. 1: Defect Detection System.

and improved BRA modules, enabling accurate detection under complex backgrounds while maintaining real-time performance.

- We construct a high-quality industrial glove surface defect dataset collected from real production lines, containing three representative defect categories with higher resolution and scene diversity than existing public datasets, which will be publicly released to support future research.
- We develop a production-oriented automatic inspection system integrating multi-camera acquisition, real-time detection, and automated rejection, validating the proposed framework in continuous industrial quality inspection scenarios.

II. Defect Detection System

To satisfy the quality inspection requirements of a continuous flexible-glove production line, a system architecture is designed as shown in Fig. 1. The system includes image acquisition, algorithm processing, rejection control, and packaging modules, which communicate through industrial interfaces to ensure continuous and real-time inspection.

During image acquisition, four industrial cameras are deployed along the production line. Camera 1 captures gloves on the mold, and Camera 2 monitors gloves at the opening position. After demolding, Cameras 3 and 4 capture the front and back views. The system adjusts frame rate and imaging parameters according to production speed and lighting conditions to maintain image clarity.

Captured images are sent to the upper computer for processing. The FG-YOLO model performs real-time defect detection, while an additional classification algorithm processes images from Cameras 1 and 2. After detection, control signals trigger the rejection unit to remove defective gloves, while qualified products continue along the line without interrupting production.

Qualified gloves then enter the packaging module, where automated equipment performs sorting, counting, and packing, reducing manual intervention and improving overall production-line automation.

III. Datasets

The quality and diversity of training data have a significant impact on the performance of defect detection models. In this study, two public glove defect datasets, GloveV2 and Glove Detect, as well as a custom dataset, are used for comparative experiments.

(a) GloveV2. (b) GloveDetect.

Fig. 2: Examples of public glove defect datasets.

A. The GloveV2 dataset

Fig. 2a shows the GloveV2 dataset, which contains five defect categories and uses dark and bright backgrounds to increase diversity. The targets include intact gloves and gloves with damage or stains. However, the dataset is limited in scale, and the backgrounds do not fully reflect real industrial environments. Many samples show severe defects, which provides limited benefit for improving detection of subtle stains or fine cracks.

B. The Glove detect dataset

Fig. 2b presents the Glove Detect dataset, which provides accurately annotated glove defects under different poses and supports model training. Its lighting conditions are close to real scenarios, and the white background reduces interference and facilitates feature extraction. However, it does not fully match industrial conditions, limiting its applicability. The dataset includes only a few defect categories, mainly common defects such as missing fingers or tearing, while rare defects are poorly represented. It also lacks sufficient normal glove samples, making it difficult for models to clearly distinguish qualified products from defective ones.

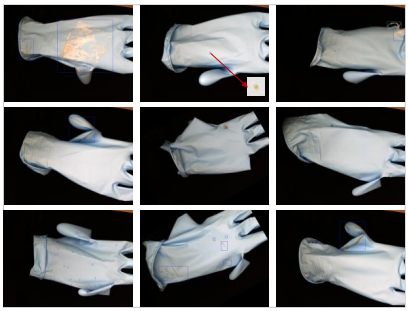


Fig. 3: Examples of oil stain defects.

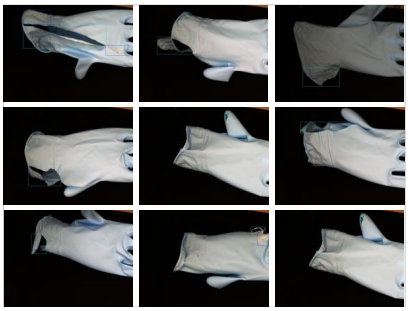


Fig. 4: Examples of damage defects.

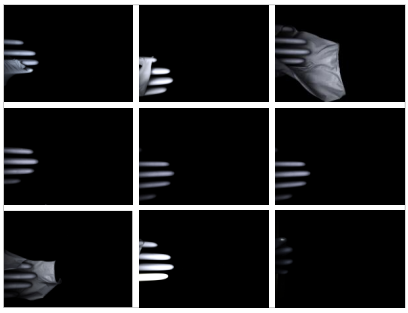


Fig. 5: Examples of unmolded defects.

C. Custom dataset

Collection and Annotation Method. To improve model generalization and reliability, a dedicated glove defect dataset is constructed using industrial cameras on a real production line. The dataset includes nitrile, latex, and PVC gloves, and images are captured on site to match actual operating conditions. Each glove is imaged against a black opaque background to reduce interference. Due to strict manufacturing standards, defective samples are rare. From about 10,000 images, several hundred defect candidates are screened and verified by industrial experts, forming a high-quality dataset. Defect categories include oil stains, damage, and unmolded gloves. Bounding boxes are annotated using a labeling tool, and YOLO-format labels are generated [11]. A double-review process is adopted, where two annotators label images independently and a third reviewer checks consistency. Images are relabeled if necessary. The final dataset contains approximately 8,000 images.

Dataset Characteristics. The dataset includes three defect categories: oil stains(Fig. 3), damage(Fig. 4), and unmolded(Fig. 5) gloves. Oil stain detection is the most challenging because stains usually appear as small, low-visibility regions and may spread across the surface, which often leads to missed or false detections. They frequently occur in folds or small areas and are easily confused with normal textures or background patterns. Damage defects are generally easier to detect but still challenging due to large variation in damage types. Simple cracks and holes are easier to recognize, while tearing and abrasion exhibit complex textures and often cause detection errors. Unmolded defects are comparatively easier to identify because the glove remains attached to the mold and presents consistent shape and clear visual characteristics. Although the datasets differ in design, Table I and Table II shows that the custom dataset contains more images and higher resolution than the others.

TABLE I: Statistics of Datasets

Dataset	Images	Defect Types	Resolution	Format
GloveV2	612	5	416×416	JPG
Glove-Detect	93	3	416×416	JPG
Ours	8000	3	1024×768	JPG

TABLE II: Statistics of Defect Categories in Our Dataset

Category	Number of Images	Resolution	Format
Oil Stain	3200	1024×768	JPG
Damaged	2400	1024×768	JPG
Unmolded	2400	1024×768	JPG

IV. Method

A. YOLO11 object detection algorithm

This work adopts YOLO11n [3] as the baseline model for improvement. It improves reliability and inference speed compared with earlier versions and supports object detection, tracking, segmentation, classification, and pose estimation. YOLO11 replaces the C2F module in YOLOv8 [3] with a smaller C3K2 module, which reduces parameter size and increases speed. The new C2PSA module employs multi-head attention to better localize small or occluded targets in complex scenes. The detection head adopts depthwise separable convolution to reduce computational cost while maintaining accuracy.

B. FG-YOLO network architecture

To address the characteristics of flexible glove defects, this paper proposes FG-YOLO (Fig. 6), which consists of a backbone, a neck, and a head. The backbone extracts image features by stacking Conv and C3K2 blocks. Each Conv block includes a convolution layer, batch normalization, and a SiLU activation function, which helps the model capture edges, textures, and patterns. The C3K2 module integrates multi-level information through grouped convolution and residual connections, preserving fine details and preventing the loss of important features. The SFC and BRA [19] modules are

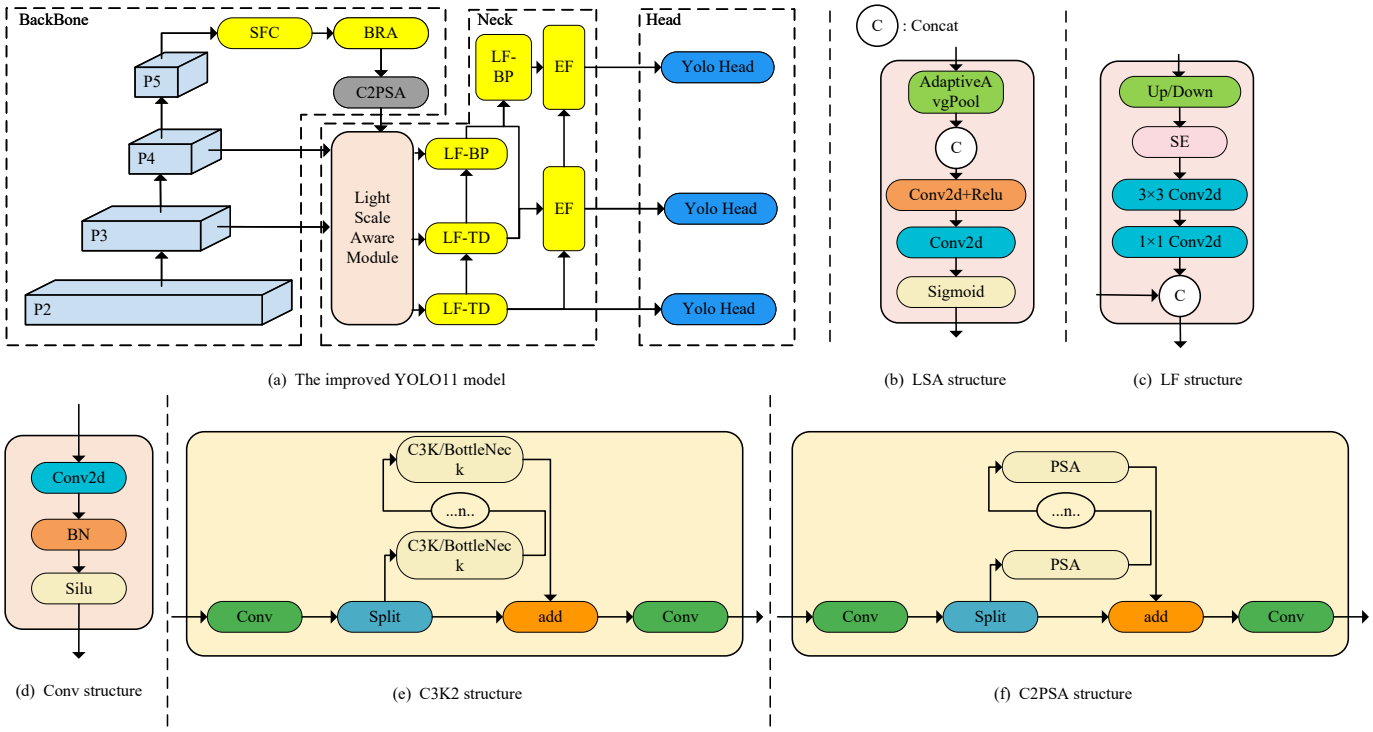


Fig. 6: Overview of FG-YOLO framework.

redesigned to strengthen feature extraction, maintain model efficiency, and enhance focus on key regions. The neck employs the proposed MSCANet to fuse multi-scale features, where the Lightweight Scale Awareness(LSA) and Lightweight Fusion (LF) modules enhance scale awareness and enable efficient cross-scale feature interaction. An Enhanced Fusion (EF) module further refines features at each scale using parallel convolutions and channel attention to highlight important information. The detection head contains three branches, where standard 3×3 convolutions are replaced with 1×1 depthwise separable convolutions to reduce computational cost.

C. SFC module

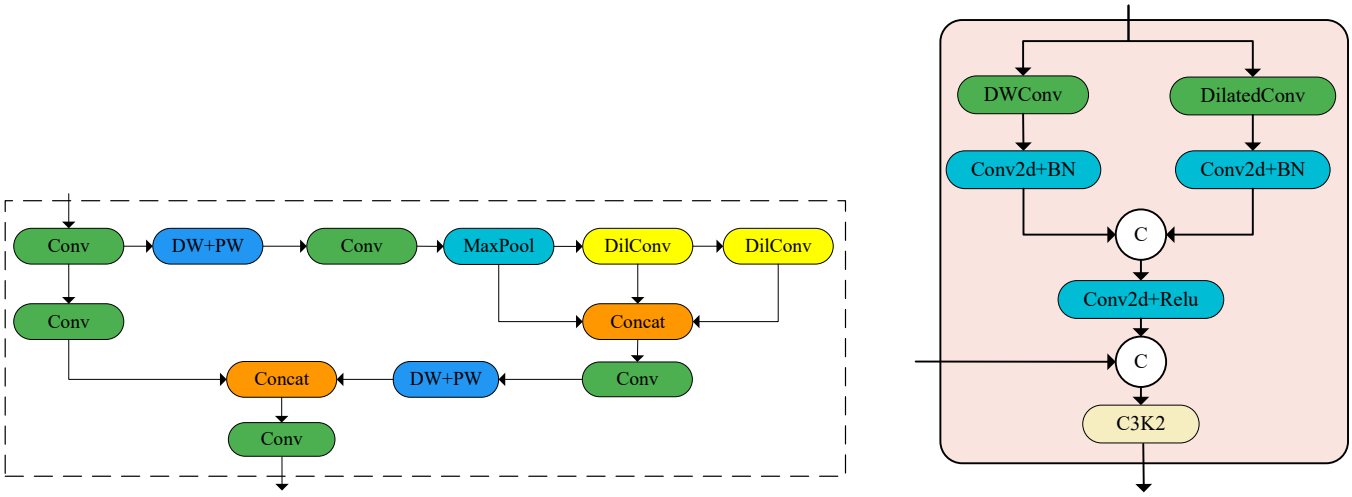
Fig. 7a illustrates the proposed SFC module, which replaces the original SPPF module in the backbone network. Although SPPF-based structures can efficiently aggregate multi-scale features, their reliance on max-pooling operations often causes the loss of fine-grained spatial details, which is unfavorable for small and low-contrast defect detection.

To alleviate this issue, the SFC module is redesigned based on the SimCSPSPPF structure by replacing standard convolutions with depthwise separable convolutions to reduce computational cost, and substituting part of the max-pooling branches with dilated convolutions to enlarge the receptive field without downsampling. This design better preserves local texture information while maintaining efficiency, resulting in more discriminative features for small-scale glove defects.

D. Improved BRA

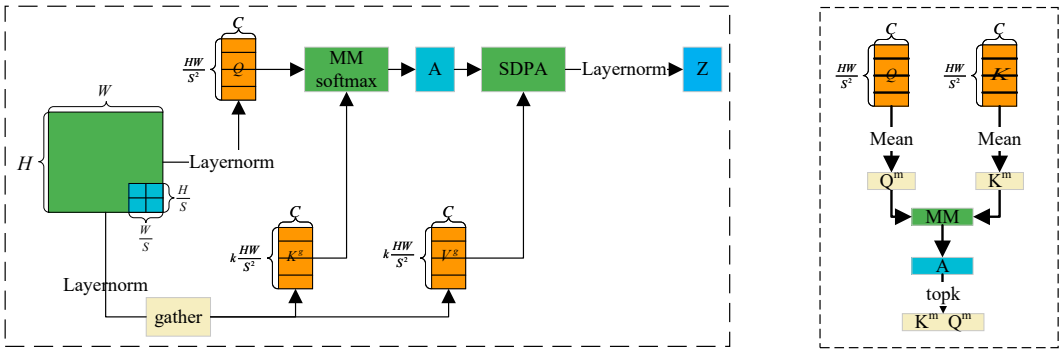
In conventional deep neural networks, feature extraction is often enhanced by stacking layers. However, increasing network depth can introduce redundant information and repeated computation, which reduces efficiency and may affect accuracy. This structure also has limited ability to capture fine details of small or blurred targets. Complex backgrounds may further interfere with feature extraction and increase the risk of false detections. To enhance feature representation in target regions while meeting real-time industrial requirements, the BRA [19] module is redesigned, as shown in Fig. 8. The original bilayer attention structure is retained, while region partition and attention computation are simplified. The input feature map of size $H \times W \times C$ is divided into non-overlapping S×S regions. Layer normalization is applied within each region to stabilize feature projection. Linear layers are then used to generate Query, Key, and Value vectors for each block to preserve token-level information. Within each block, the Query and Key vectors are averaged along the token dimension to obtain a regional correlation matrix. This matrix measures similarity among blocks. For each block, the indices corresponding to the top-k values in A^r are selected to identify the most relevant regions. This strategy reduces computational complexity while filtering out irrelevant areas. After the relevant regions are determined, the corresponding features are gathered from the original Key and Value matrices and aggregated into K^g and V^g while the Query matrix remains unchanged.

This work also adopts acceleration techniques such as



(a) Overview of SFC framework. (b) Structure of EF module.

Fig. 7: Architecture of the proposed method.



(a) Improved BRA module (b) Gather module

Fig. 8: Improved BRA module structure.

Scaled Dot Product Attention (SDPA) in PyTorch and Flash Attention [20] to improve computational speed and memory efficiency. To enhance spatial localization awareness, a depth-wise separable convolution (DWConv) is directly applied to the original Value branch to extract fine-grained positional features. The DWConv output is added to the attention output, and the result is then combined with the original block features through a residual connection, followed by layer normalization. Finally, all processed blocks are reconstructed into an H×W×C feature map through an inverse partition operation. The related formulas are presented as follows:

$$X_{\text{norm}} = \text{Layernorm}(X) \tag{1}$$

$$A^r = Q^m (K^m)^T \tag{2}$$

$$K^g = \text{gather}(K, \text{top}(A^r)) \tag{3}$$

$$V^g = \text{gather}(V, \text{top}(A^r)) \tag{4}$$

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{C}}\right) V \tag{5}$$

$$Z = \text{Layernorm}(\text{Attention}(Q, K^g, V^g) + \text{DWConv}(V)) \tag{6}$$

E. Multi-Scale Cross Attention Network

To address the limited cross-scale interaction and insufficient detail preservation in traditional feature pyramid structures, a Multi-Scale Cross Attention Network (MSCANet) is proposed, as shown in Fig. 6 and Fig. 7b. MSCANet introduces lightweight scale-aware weighting and efficient cross-scale attention to enhance multi-scale feature fusion while maintaining low computational cost, thereby improving the perception of small and fine-grained defects.

MSCANet takes three-level backbone feature maps as input and consists of three core modules: LSA, LF, and EF. The LSA module adaptively reweights multi-scale features by estimating scale importance from global contextual information, enabling the network to emphasize informative scales for defect detection. The LF module facilitates bidirectional feature interaction through a top-down (LF-TD) and a bottom-up (LF-BP) path, where LF-TD propagates high-level semantic information to shallow layers to compensate for detail loss, and LF-BP injects geometric details into deeper layers to enhance semantic consistency across scales. The EF module further refines the fused features using a dual-branch structure, where depthwise separable convolution focuses on local detail extraction and

TABLE III: Performance Comparison of Different Detection Algorithms

Model	Params (M)	Precision (%)	Recall (%)	mAP50 (%)	mAP50:95 (%)	FPS
YOLOv5n[21]	2.5	84.97	80.06	84.17	58.69	**312**
YOLOv3-tiny[10]	2.1	84.52	82.50	84.73	58.42	280
YOLOv8n[3]	3.2	84.43	82.65	85.73	58.38	285
YOLOv10n[4]	3.6	85.30	84.20	84.81	58.22	290
YOLO11n[3]	2.6	80.22	85.93	84.65	58.30	294
YOLO12n[22]	3.9	85.87	85.54	84.23	58.02	253
RT-DETR-R50[6]	42.0	**87.81**	72.59	84.84	54.00	33
FDM-YOLO[23]	4.1	85.82	84.32	85.90	58.80	260
FG-YOLO	4.2	87.80	**87.43**	**89.54**	**60.59**	280

TABLE IV: Comparison of Test Results on Multiple Datasets

Dataset	Method	Precision (%)	Recall (%)	mAP50 (%)
GloveV2	YOLO11n	**83.15**	82.81	87.38
	FG-YOLO	82.77	**87.92**	**92.22**
Glove-Detect	YOLO11n	70.10	58.13	62.62
	FG-YOLO	**76.67**	**58.78**	**64.94**
VisDrone2019	YOLO11n	43.32	29.61	30.70
	FG-YOLO	**44.50**	**33.02**	**33.38**
Custom	YOLO11n	80.22	85.93	84.65
	FG-YOLO	**87.80**	**87.43**	**89.54**

dilated convolution expands the receptive field to capture broader contextual information. The outputs of both branches are aggregated with channel attention and combined with the LF output via a skip connection, ensuring complementary integration of semantic and fine-grained information.

V. Experiment

A. Experimental Environment and Evaluation Metrics

Model training is conducted on an Ubuntu 22.04.5 LTS system. The hardware configuration includes a 13th-generation Intel Core i9-13900K CPU, an NVIDIA GeForce RTX 4090 GPU with 24 GB of memory, and 128 GB of RAM. The deep learning framework is PyTorch 2.0.1 with CUDA 12.6 and cuDNN 11.7. All inference tests are performed on an NVIDIA GeForce GTX 1070Ti GPU, and all the settings were the same.

Training and testing are conducted on the custom glove defect dataset, while two public glove defect datasets and VisDrone2019 are used for comparative evaluation. The dataset is split into training, validation, and test sets with a 7:2:1 ratio. Data augmentation, including sample balancing and Mosaic augmentation, is applied to improve robustness. The model is optimized using Adam and trained for up to 500 epochs with early stopping. Input images are resized to 640×640 pixels to balance GPU memory usage and resolution.

Model performance is evaluated using Precision, Recall, mAP50, mAP50:95, and FPS. mAP50 and mAP50:95 represent detection accuracy under single and multiple Intersection over Union (IoU) thresholds, respectively, while FPS measures inference speed. The formulas for AP and mAP are given as follows:

$$AP = \int_0^1 P(R)\, dR \tag{7}$$

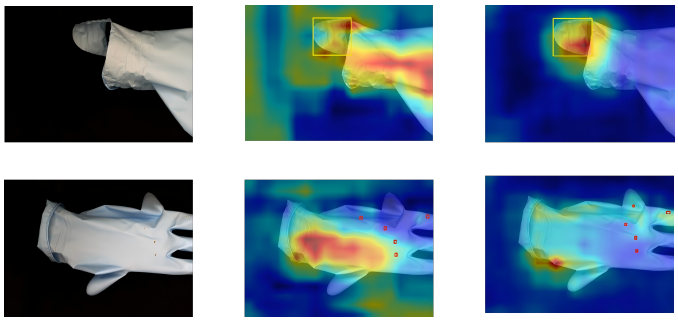


Fig. 9: Heatmap comparison.

$$mAP = \frac{1}{N} \sum_{i=1}^{N} AP_i \tag{8}$$

B. Comparative Experiments

Fig. 9 compares heatmaps before and after integrating the redesigned BRA module, and Table III reports quantitative comparisons with mainstream detectors. FG-YOLO achieves 87.80% precision, comparable to RT-DETR-R50 and higher than all YOLO variants, while also obtaining the highest recall of 87.43%, indicating fewer missed detections. It further reaches 89.54% mAP50 and 60.59% mAP50:95, outperforming all compared models across IoU thresholds and demonstrating improved robustness and generalization. Although its inference speed of 280 FPS is slightly lower than some YOLO variants, it remains much faster than RT-DETR-R50, showing that the additional computation remains acceptable. Cross-dataset evaluation in Table IV further confirms strong generalization, where FG-YOLO consistently surpasses YOLO11n, with a notable +3.84% mAP improvement on

TABLE V: Algorithm ablation experiment

MSCANet	Improved BRA	SFC	Precision (%)	Recall (%)	mAP50 (%)	mAP50:95 (%)
×	×	×	80.22	85.93	84.65	58.30
✓	×	×	**89.22**	86.07	87.87	58.82
×	✓	×	82.61	84.25	85.63	59.36
×	×	✓	83.81	82.45	85.95	60.00
✓	✓	×	87.72	85.54	87.91	60.43
×	✓	✓	88.94	84.89	87.65	59.08
✓	×	✓	85.74	85.24	85.52	60.38
✓	✓	✓	87.80	**87.43**	**89.54**	**60.59**

TABLE VI: Comparison of Attention Mechanisms

Method	Params (M)	mAP50 (%)	FPS
YOLO11n	2.6	84.65	**294**
+CAM[24]	6.3	84.78	201
+SHSA[25]	4.3	84.82	220
+CBAM[26]	4.3	85.01	216
+BRA	4.4	85.13	210
+Improved BRA	4.5	**85.63**	234

TABLE VII: Ablation Study of the SFC Module

Method	mAP50 (%)	FPS
YOLO11	84.65	**294**
+SimCSPSPPF	85.24	260
+DilatedConv	85.22	272
+(DwConv+PwConv)	85.35	282
+SFC	**85.95**	290

TABLE VIII: Ablation Study of MSCANet

Method	mAP50 (%)	FPS
YOLO11	84.65	**294**
+LSA	85.30	290
+LF	87.05	279
+EF	86.15	284
+MSCANet	**87.87**	274

GloveV2. A slight recall drop on Glove Detect is likely caused by its limited scale and simpler scenes. Visual comparisons in Fig. 10 also show that FG-YOLO reduces missed and false detections compared with YOLOv5n, YOLOv8n, YOLOv11n, and RT-DETR-R50, particularly for oil stains and complex damage patterns, demonstrating that improved feature fusion effectively enhances detection reliability in practical industrial scenarios.

C. Ablation Study

Ablation experiments on the YOLO11n baseline evaluate the effects of MSCANet, the SFC module, and the improved BRA module. As shown in Table V, MSCANet alone provides the largest gains in recall and mAP50, highlighting the importance of effective multi-scale feature fusion for small defect detection. The improved BRA module further increases mAP50:95, indicating enhanced feature discrimination and robustness under complex backgrounds, while the SFC module contributes to overall feature representation with minimal efficiency loss.Combining different modules produces complementary effects. Integrating MSCANet with either SFC or improved BRA significantly improves small object detection performance, while incorporating all components in FG-YOLO achieves the best overall results, reaching 89.54% mAP50 and 60.59% mAP50:95. These results demonstrate that the proposed modules jointly enhance detection accuracy while preserving real-time performance.

Additional comparisons in Table VI show that BRA achieves higher accuracy under similar parameter budgets, indicating a more favorable accuracy–efficiency trade-off. Although Improved BRA introduces a slight increase in parameters, it simplifies routing operations and adopts efficient attention implementations (e.g., SDPA), which reduces computational overhead and leads to faster inference. Table VII further confirms that the redesigned SFC module achieves a better balance between detection accuracy and inference speed.Table VIII presents the ablation results of MSCANet, showing that each individual module contributes to performance improvement, while the complete MSCANet achieves the best mAP50 with a modest reduction in inference speed, demonstrating a favorable accuracy–efficiency trade-off.

VI. Conclusion

To address missed small surface defects, strong background interference, and the trade-off between accuracy and efficiency in existing methods, this work designs and implements a flexible glove defect detection system and proposes an improved YOLO11-based detection framework, FG-YOLO. A redesigned SFC module is introduced to enhance feature representation, an improved BRA module is embedded to suppress background noise and redundant information, and MSCANet is proposed to enable adaptive feature enhancement and efficient cross-scale fusion. A dedicated industrial glove defect dataset with three representative defect categories is constructed, and extensive comparative and ablation experiments verify the effectiveness of the proposed approach. Experimental results demonstrate that FG-YOLO satisfies real-time inspection requirements in industrial production environments. Future work will focus on expanding dataset diversity and exploring further model lightweighting strategies to improve generalization and support deployment on resource-constrained edge devices.

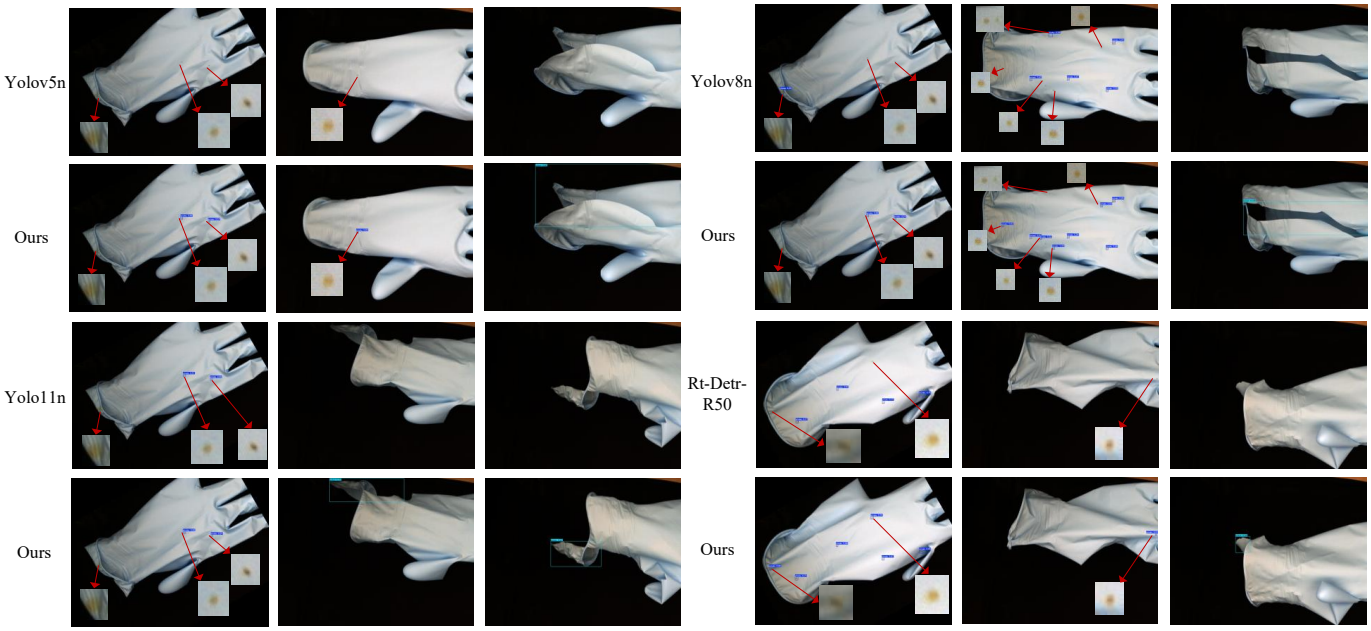


Fig. 10: Comparison of visual defect detection using different algorithms.

REFERENCES

[1] S. M. Ali, S. Noghanian, Z. U. Khan, and M. R. Islam, "Wearable and flexible sensor devices: Recent advances in designs, fabrication methods, and applications," *Sensors*, vol. 25, no. 5, Art. no. 1377, 2025.

[2] L. Chai, L. Ren, K. Gu, and Y. Zhang, "Vision sensing based intelligent detection of surface defect and its industrial applications," *Comput. Integr. Manuf. Syst.*, vol. 28, no. 7, pp. 1996–2004, 2022.

[3] R. Sapkota and M. Karkee, "Ultralytics YOLO evolution: An overview of YOLO object detectors," arXiv preprint arXiv:2510.09653, 2025.

[4] A. Wang, H. Chen, L. Liu, K. Chen, Z. Lin, and J. Han, "YOLOv10: Real-time end-to-end object detection," in Proc. Adv. Neural Inf. Process. Syst. (NeurIPS), 2024.

[5] D. Reis, J. Kupec, J. Hong, and S. Daudt, "Real-time flying object detection with YOLOv8," arXiv preprint arXiv:2305.09972, 2023.

[6] Y. Zhao, W. Lv, S. Xu, J. Wei, Y. Wang, and X. Li, "DETRs beat YOLOs on real-time object detection," in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), 2024, pp. 16965–16974.

[7] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo, "Swin transformer: Hierarchical vision transformer using shifted windows," *Int. J. Comput. Vis.*, vol. 131, pp. 100–119, 2023.

[8] X. Zhu, W. Su, L. Lu, B. Li, X. Wang, and J. Dai, "Deformable DETR: Deformable transformers for end-to-end object detection," *Int. J. Comput. Vis.*, vol. 131, pp. 267–289, 2023.

[9] H. Zhang, P. Xiao, F. Yao, Q. Zhang, and Y. Gong, "Fusion of multi-scale attention for aerial images small-target detection model based on PARE-YOLO," *Sci. Rep.*, vol. 15, no. 1, Art. no. 4753, 2025.

[10] F. Feng, L. Yang, Q. Zhou, and H. Wang, "YOLO-tiny: A lightweight small object detection algorithm for UAV aerial imagery," *IET Image Process.*, vol. 19, no. 1, 2025.

[11] H. Jin, R. Du, L. Qiao, and X. Zhang, "CCA-YOLO: An improved glove defect detection algorithm based on YOLOv5," *Appl. Sci.*, vol. 13, no. 18, Art. no. 10173, 2023.

[12] Y. Ge, X. Zhang, and Y. Liu, "YOLO-MSD: A robust industrial surface defect detection model based on multi-scale feature fusion," *Appl. Intell.*, early access, 2025.

[13] J. H. Tian, K. Li, and Y. Zhou, "An improved YOLOv5n algorithm for detecting surface defects," *Sci. Rep.*, vol. 15, Art. no. 94109, 2025.

[14] M. Fang, X. Rui, and H. Cheng, "Small object detection algorithm based on improved attention mechanism and feature fusion of YOLOv8," *J. Adv. Comput. Intell. Intell. Informatics*, vol. 29, no. 4, pp. 941–952, 2025.

[15] R. Kühlechner, H. Huang, and S. B. Kang, "Object detection survey for industrial applications with focus on surface defects," *J. Intell. Manuf.*, early access, 2025.

[16] G. Liu, "Surface defect detection methods based on deep learning: A brief review," in *Proc. 2020 2nd Int. Conf. Inf. Technol. Comput. Appl. (ITCA)*, 2020, pp. 200–203, doi: 10.1109/ITCA52113.2020.00049.

[17] Y. Ma, J. Yin, F. Huang, and. Q Li, "Surface defect inspection of industrial products with object detection deep networks: A systematic review," *Knowl.-Based Syst.*, vol. 284, Art. no. 111356, 2024.

[18] M. Nikouei, B. Baroutian, and S. Nabavi, "Small object detection: A comprehensive survey on challenges, techniques and real-world applications," arXiv preprint arXiv:2503.20516, 2025.

[19] L. Zhu, X. Wang, Z. Ke, W. Zhang, and Y. Lau, "BiFormer: Vision transformer with bi-level routing attention," in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), 2023, pp. 10323–10333.

[20] T. Dao, "FlashAttention-2: Faster attention with better parallelism and work partitioning," arXiv preprint arXiv:2307.08691, 2023.

[21] R. Khanam and M. Hussain, "What is YOLOv5: A deep look into the internal features of the popular object detector," arXiv preprint arXiv:2407.20892, 2024.

[22] Y. Tian, Q. Ye, and D. Doermann, "YOLOv12: Attention-centric real-time object detectors," arXiv preprint arXiv:2502.12524, 2025.

[23] X. Zhang, "A lightweight model FDM-YOLO for small target improvement based on YOLOv8," arXiv preprint arXiv:2503.04452, 2025.

[24] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba, "Learning deep features for discriminative localization," in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), 2016, pp. 2921–2929.

[25] S. Yun and Y. Ro, "ShViT: Single-head vision transformer with memory efficient macro design," in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), 2024, pp. 5756–5767.

[26] S. Woo, J. Park, J. Y. Lee, and I. S. Kweon, "CBAM: Convolutional block attention module," in Proc. Eur. Conf. Comput. Vis. (ECCV), 2018, pp. 3–19.