Computers, Materials & Continua

Tech Science Press

https://doi.org/10.32604/cmc.2026.085293

ARTICLE

GSL-YOLOv8-EMA: A Lightweight Textile Surface Defect Detection Algorithm Based on Improved YOLOv8

Jiaji Pan, Yanxin Xiang, Fujun Liu and Wei Wang*

College of Engineering and Design, Hunan Normal University, Changsha, China

*Corresponding Author: Wei Wang. Email: weeedesign@hunnu.edu.cn

Received: 08 May 2026; Accepted: 26 August 2026; Published: 22 October 2026

ABSTRACT: Inspecting surface defects on hook-and-loop fasteners is tricky due to background texture interference and sub-millimeter flaw scales. To tackle this, we present GSL-YOLOv8-EMA, a lightweight detector tailored for real-time industrial production. We first employed Partial Convolution (PConv) to replace standard convolution in the backbone, significantly cutting computational overhead. In the neck, we designed a Slim-neck architecture to strike a balance between lightweight design and feature extraction capability. To recover microscopic details lost in feature maps, we replaced standard upsampling with the CARAFE operator and integrated an EMA module to suppress background noise. Combined with the WIoU loss to refine boundary localization for thin defects, our model achieves strong accuracy while maintaining high inference speed. Experimental tests on our dataset demonstrate strong performance. The GSL-YOLOv8-EMA achieves an mAP@0.5 of 88.2 ± 3.1%, marking a 9.8% increase over the baseline. Meanwhile, the model runs at a high speed of 148 frames per second (FPS). These results demonstrate that our model is both accurate and lightweight. It is well-suited for real-time defect inspection in factory production lines.

KEYWORDS: Surface defect detection; textiles; YOLOv8; lightweight networks; attention mechanisms

1 Introduction

The unique two-way adhesive design of hook-and-loop fasteners is widely used in the fields of clothing, aerospace and precision manufacturing [1]. The surface quality of hook-and-loop fasteners is important for the commercial value and safety of the final product. However, with the growth of market demand for high-quality textiles, traditional quality control methods are difficult to keep up with the pace of modern production. At present, the surface defect detection of hook-and-loop fasteners mainly hinges on human operators for quality checks. Besides being inefficient, this method relies heavily on operator subjectivity, resulting in frequent missed detection or false detection. On the industrial production line, if minor surface defects cannot be eliminated in time, it will not only cause significant economic losses, but also damage the company's brand reputation. Therefore, overcoming the limitations of manual detection and realizing high-precision and high-efficiency automatic detection has become an urgent requirement for the transformation and upgrading of the textile industry.

The rapid development of computer vision and artificial intelligence has introduced new solutions for textile defect detection. Compared with traditional physical detection, intelligent visual detection based on deep learning has significant advantages, including non-contact operation, high speed, and continuous function. It automatically extracts features from massive datasets to accurately identify defects under complex texture background [2,3]. The integration of AI into the textile industry is not only important for

Copyright © 2026 The Authors. Published by Tech Science Press. This work is licensed under a Creative Commons Attribution 4.0 International License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

replacing artificial vision detection and reducing labor costs, but also can be used as the core technology basis for improving the automation of production lines and enhancing the competitiveness of enterprises.

Although fabric defect detection has been extensively studied, the existing technology still faces multiple challenges when dealing with the complex texture structure and micro defect morphology of hook-and-loop fasteners.

In recent years, the research of textile defect detection has rapidly developed from traditional manual features to deep learning and self-supervised methods [4]. Early studies mainly relied on filtering and statistical features, such as Gabor [5,6], Local Binary Pattern (LBP) [7,8], and Gray-Level Co-occurrence Matrix (GLCM) [9,10], often paired with Support Vector Machines (SVM) [11]. Although these methods are easy to implement and provide good real-time performance, they show poor robustness to light changes, fabric batch differences, and complex textures, which makes them not suitable for highly variable fabric textures in modern industry.

In order to solve the practical production problems, many studies have focused on the specific technical bottlenecks. Real-time target detection based on the YOLO series has become the mainstream choice in the industry [12,13]. Addressing network latency constraints, Chen et al. [14] proposed Partial Convolution (PConv) and developed a new neural network structure named FasterNet. This architecture achieves higher execution speed while maintaining excellent accuracy in a variety of visual tasks. Concurrently, resolving the frequent omission of small-scale or multi-scale defects has driven the integration of attention mechanisms [15] and multi-scale feature fusion strategies [16]. Zhao et al. [17] proposed Attention-Based Multiscale Feature Fusion (AMFF), which is an attention multi-scale feature fusion module with self-enhancement and cross-enhancement mechanisms. By enhancing defect features, AMFF significantly improves the accuracy of industrial defect detection while ensuring real-time performance. Regarding complex printed or jacquard backgrounds, Li et al. [18] proposed Fisher criterion-based stacked denoising autoencoders (FCSDA), a stacked denoising autoencoder method based on the Fisher criterion. Through image block classification and residual reconstruction, under the condition of limited negative samples, the defect detection and localization in complex pattern fabrics are effectively realized. Traditional upsampling methods rely only on spatial location and lack awareness of image semantic content. In contrast, the lightweight content-aware reassembly operator CARAFE [19] significantly improves the model performance in the feature fusion stage. To extend and generalize the CARAFE mechanism, Wang et al. [20] proposed CARAFE++, enabling simultaneous support for content-aware upsampling and downsampling. Its unified dynamic feature reorganization mechanism shows superior performance in multi-scale feature fusion. Traditional bounding box regression loss often over-penalizes low-quality samples, resulting in harmful gradient oscillations during training. Additionally, mitigating gradient oscillations caused by low-quality bounding boxes during training, the Wise-IoU (WIoU) loss function [21] was developed. It utilizes a dynamic focusing mechanism to improve overall model generalization.

To meet these operational and algorithmic challenges, this paper focuses on the surface defect detection of the hook-and-loop in the industrial scene, and proposes a lightweight detection scheme GSL-YOLOv8-EMA based on improved YOLOv8. At the data level, the industrial image acquisition platform was constructed independently, and a special dataset containing a variety of typical defects was constructed. At the algorithmic level, in order to solve the difficulties of real-time processing and micro-target detection, we structurally optimized YOLOv8. We integrated FasterNet [14] as a lightweight backbone to reduce computational load. We then incorporated the CARAFE [19] operator to enhance semantic upsampling and embedded a multi-scale attention module [22] to actively suppress background noise. Finally, the WIoU [21] was applied to refine the localization precision of elongated flaws. At the validation level, the robustness of the algorithm was verified through ablation studies and horizontal comparisons on public datasets such as

Northeastern University Surface Defect Database (NEU-DET) and Textile Texture Database (TILDA). The following chapters of this paper will introduce the theoretical basis of deep learning, the construction of the hardware platform and data processing, the specific derivation of algorithm improvement and the analysis of experimental results in detail.

The paper is arranged as follows: Section 2 covers data acquisition and processing. Section 3 introduces the GSL-YOLOv8-EMA architecture. Section 4 presents the experimental design and comparative results. Section 5 discusses the model's performance and current limits, and Section 6 summarizes the work.

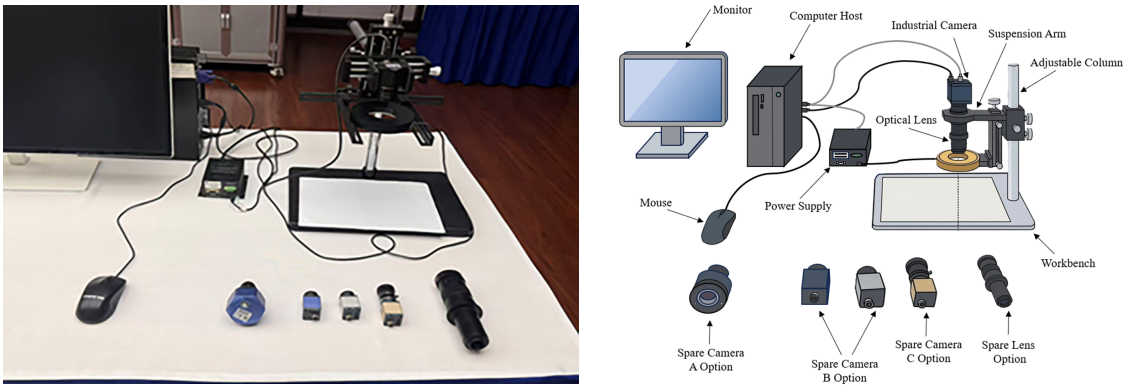
2 Dataset Construction and Preprocessing

High-quality datasets are a prerequisite for training robust deep learning models. In view of the lack of publicly available high-resolution datasets specifically used for the defect detection of hook-and-loop fasteners, this section establishes a special industrial vision acquisition platform and proposes a comprehensive dataset construction framework, including the process from data annotation to data expansion.

2.1 Data Acquisition System

In order to suppress the complex background noise and meet the requirements of high-speed and high-precision detection of industrial production lines, a customized image acquisition system is designed.

Considering the high-speed motion of hook-and-loop fasteners on conveyor belts, the system employs a global shutter industrial area scan camera (Basler acA2500-14gc) equipped with a high-sensitivity Complementary Metal–Oxide–Semiconductor (CMOS) sensor. This selection effectively prevents the motion blur typically generated by rolling shutters. The camera resolution is configured to 5 megapixels. The imaging system achieves a 100–150 mm FOV and 0.05 mm resolution. We used a 16 mm prime lens with a 5 mm extension tube and an LED ring light. The ring light reduces shadow noise from the background texture, making surface defects clearly visible. Fig. 1 shows the complete image acquisition system.



(**a**) Photograph of the actual setup (**b**) Schematic diagram of the apparatus

Figure 1: Schematic diagram of the industrial data acquisition system.

To mirror real factory setups, we introduced lighting variations and slight spatial movements like translation and rotation during image capture. After data collection, the invalid overexposed samples were manually removed, and finally 102 representative high-quality and high-resolution original defect images were selected.

2.2 Defect Definition and Annotation Scheme

Based on practical production requirements and the visual characteristics of the samples, the dataset categorizes the defects on hook-and-loop fasteners into two core target classes, as shown in Fig. 2.

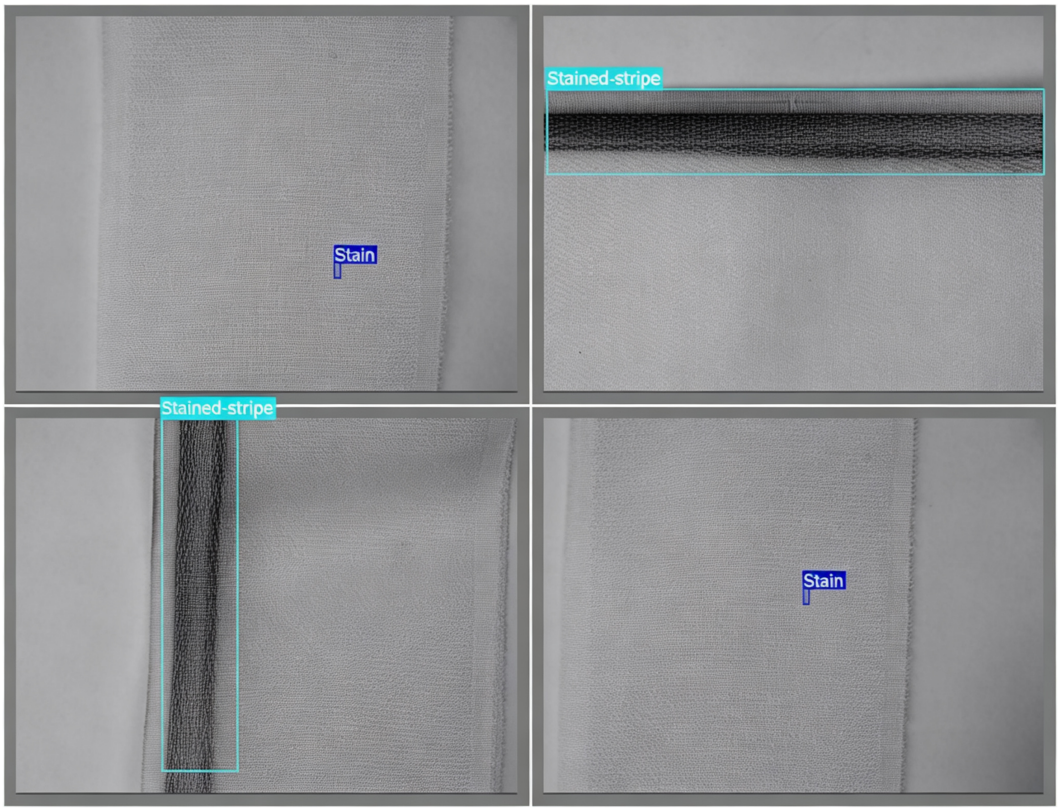


Figure 2: Annotation examples for the two types of defects.

Stain: Oil spots or dust clumps characterized by irregular morphologies, small areas, and dark appearances.

Stained-stripe: Streak-like contaminations caused by friction or liquid leakage, exhibiting extreme aspect ratios, lighter colors, and large spatial spans.

The LabelImg tool was employed to generate normalized coordinate annotation files in strict accordance with the standard output format of the YOLO algorithm. To augment feature learning for minute objects and complex morphologies, the bounding boxes were required to tightly enclose the defect edges, thereby maximizing the target signal-to-noise ratio.

2.3 Combinatorial Data Augmentation Strategy

Given the small initial sample size, it is very easy to use it directly for deep network training. To solve this problem, a strategy combining "offline expansion and online enhancement" is designed to enrich the semantic space of samples and boost its generalization performance.

Offline data expansion built on physical disturbance: geometric transformation: simulate the position uncertainty caused by the vibration and deviation of the production line by applying random rotation, translation and flipping. This enables the network to learn deep features with translation and rotation

invariance. Photometric distortion: in HSV and RGB color spaces, contrast, brightness and saturation are randomly disturbed to simulate the aging of the light source and the fluctuation of material reflectivity, so as to boost the robustness of the network to absolute pixel values. Noise and blur injection: superimpose Gaussian noise to simulate sensor circuit interference, and apply a blur operator to simulate optical defocus and motion blur. This hybrid strategy aims to destroy local redundant textures and direct the model's attention toward a more robust global topology. Through offline enhancement, the dataset is expanded to 500 images, which alleviates the problem of sparse features in small sample scenes.

Mosaic-based online data expansion: in the iterative stage of network training, the mosaic expansion algorithm is introduced. The strategy randomly selects four images, scales and cuts them, and then stitches them into a high-resolution input image [23]. This operation not only multiplies the background complexity of the training context, but also effectively strengthens detection efficacy for stains and stripes due to its inherent multiscale characteristics. In addition, the fusion of multiple samples enhances the balance of positive sample distribution in a single batch, thus optimizing the statistical properties of the batch normalization layer.

2.4 Dataset Statistics and Partitioning

Aiming to verify the validity of the dataset distribution and the detection model efficacy, the expanded dataset was statistically analyzed. In terms of category distribution, the number of instances of staining and staining stripes is balanced, which effectively avoids the category deviation problem usually caused by long tail distribution. With regard to the spatial distribution and geometric scale, the center point of the defect presents a global random and discrete distribution, and the aspect ratio of the target spans a very wide range. This puts forward strict requirements for the feature fusion mechanism of detection network.

Using hierarchical sampling, 500 enhanced images are strictly divided into the training set (400 images), verification set (50 images, used to monitor the early stop mechanism and adjust the learning rate) and test set (50 images, reserved for the final performance evaluation) at the ratio of 8:1:1.

3 Methods

3.1 Overview of the GSL-YOLOv8-EMA Network Architecture

Although the baseline YOLOv8 algorithm has established a performance benchmark in general-purpose object detection, it still exhibits architectural limitations when directly applied to surface defect detection on hook-and-loop fasteners. Standard convolution operations share weights spatially and lack a perceptual bias toward regions of interest (RoI), rendering the model inadequate at effectively suppressing complex background texture interference. Furthermore, as the network depth increases, the high-frequency edge information of minute defects is highly susceptible to over-smoothing, thereby hindering the achievement of high-precision detection.

We propose GSL-YOLOv8-EMA to tackle these issues, focusing on both high precision and a lightweight footprint, as illustrated in Fig. 3.

The proposed model integrates five main modules: a lightweight C2f-Faster backbone, a Slim-neck feature fusion layer, CARAFE content-aware upsampling, an EMA mechanism, and the Wise-IoUv3 loss.

We tailored this framework for high-speed surface inspection through a few focused changes. Partial convolutions replace standard convolution, reducing memory access overhead and spatial redundancy. The Slim-neck architecture combines GSConv and VoVGSCSP, making the model substantially more lightweight. Standard upsampling often misses small stains, so we use the CARAFE operator to dynamically generate kernels that recover fine defect details from textured backgrounds. To suppress background noise from

hook patterns, the EMA module captures multi-scale and cross-spatial features with minimal compute cost. Lastly, we use WIoUv3 as the loss function to handle varying defect sizes and noisy annotations. Its dynamic focusing mechanism assigns gradient weights based on sample quality, preventing the model from overfitting to blurry target data.

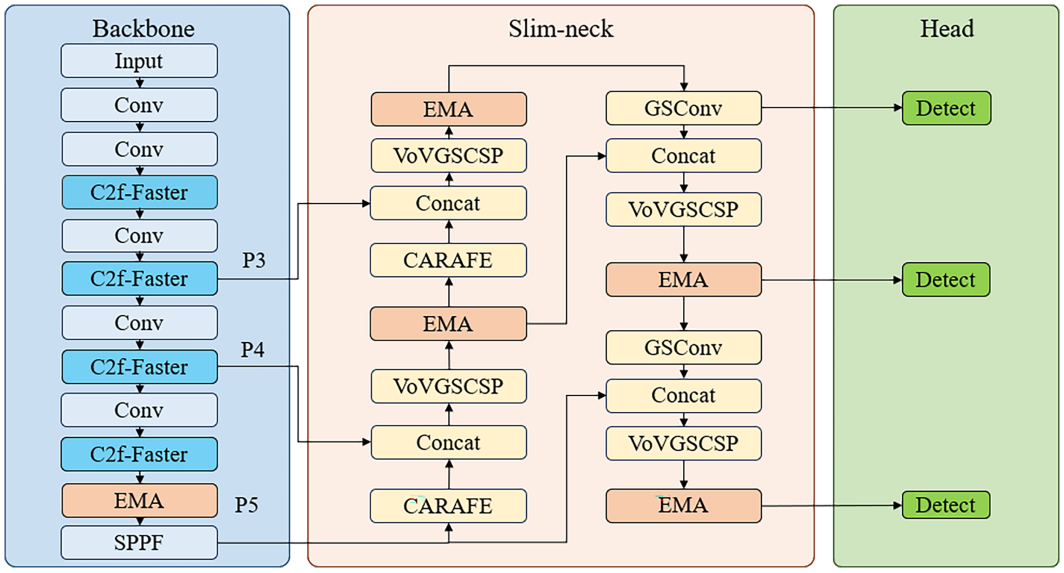


Figure 3: Schematic diagram of the GSL-YOLOv8-EMA architecture.

The nomenclature of the GSL-YOLOv8-EMA algorithm is defined as follows: G represents the GSConv-based lightweight neck; S stands for the Structure-aware CARAFE upsampling; L denotes the Lightweight backbone; EMA indicates the Efficient Multi-scale Attention mechanism.

3.2 Lightweight Reconstruction of the Backbone Network: C2f-Faster

To alleviate the extensive computational redundancy prevalent in standard convolutional neural networks, the C2f in YOLOv8n is exchanged for C2f-Faster. The core of C2f-Faster is replacing standard convolution with the Partial Convolution (PConv). PConv applies spatial convolution to only a quarter of the input feature channels while leaving the remaining three-quarters of the channels untouched. Subsequently, pointwise convolution (PWConv) is applied to fuse channel information, as illustrated in Fig. 4.

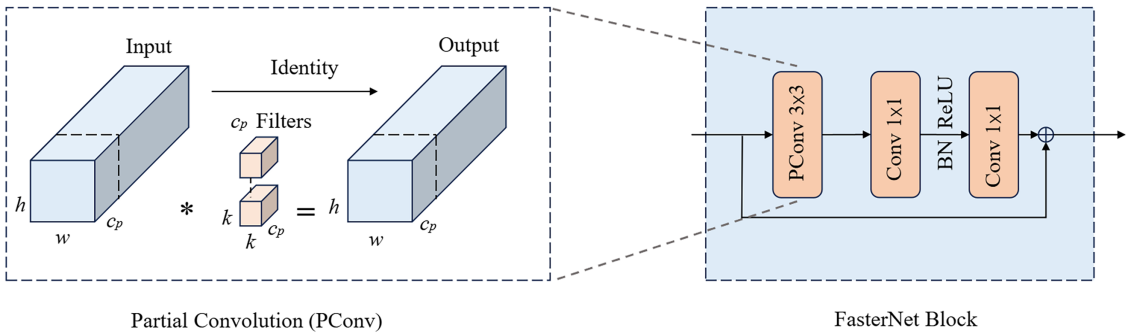


Figure 4: Schematic diagram of the PConv structure.

Assuming the spatial dimensions of the input feature map are $h \times w$, the channel count of is c, the convolution kernel size is k, the FLOPs of PConv are calculated as follows:

$$FLOPs_{PConv} = h \times w \times k^2 \times c_p^2 = \frac{1}{16} FLOPs_{Conv} \tag{1}$$

where c_p denotes the channel dimension participating during convolution. As a result of this modification, we bring the computational load down to about 1/16 of what a standard convolution requires. This allows us to significantly reduce memory access latency and processing complexity, all while fully preserving the extraction of complex texture features.

3.3 Slim-Neck Design Based on GSConv and VoVGSCSP

Standard convolutions (SC) become too heavy when channel numbers grow during multi-scale fusion in the neck. Depthwise separable convolutions (DSC) cut the compute cost, but they tend to miss faint stains. We use a Slim-neck architecture with GSConv to fix this trade-off, as shown in Fig. 5.

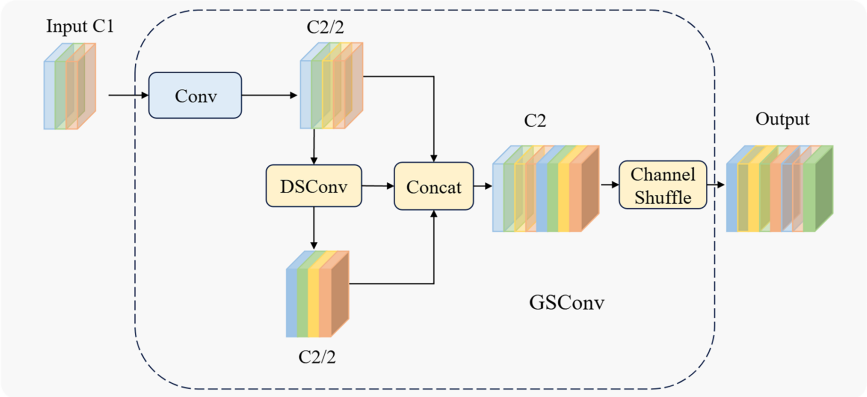


Figure 5: Schematic diagram of the GSConv structure.

GSConv combines the texture preservation of SC with the low parameter count of DSC. It passes input features through a channel-halving SC layer and then a DSC layer. Concatenating these outputs with a channel shuffle mixes information across all channels. The output X_{out} can be expressed as:

$$X_{out} = Shuffle\left(Concat\left(X_{SC}, X_{DSC}\right)\right) \tag{2}$$

where X_{SC} and X_{DSC} represent the outputs of standard and depthwise separable convolutions, respectively.

The neck network uses VoVGSCSP instead of the standard C2f module, as shown in Fig. 6. By linking GSConv layers and combining features in one step, it lowers inference latency and avoids gradient degradation.

3.4 Content-Aware Feature Upsampling Based on CARAFE

YOLOv8 defaults to nearest-neighbor upsampling. The method simply duplicates pixels. This approach is highly prone to distorting the geometric morphologies of elongated defects and introducing background noise. Unlike traditional upsampling methods that rely on fixed kernels, the CARAFE dynamically generates instance-specific upsampling kernels based on the local content of the input features [19]. To preserve morphological integrity, the CARAFE module is embedded in the proposed framework, as illustrated in Fig. 7.

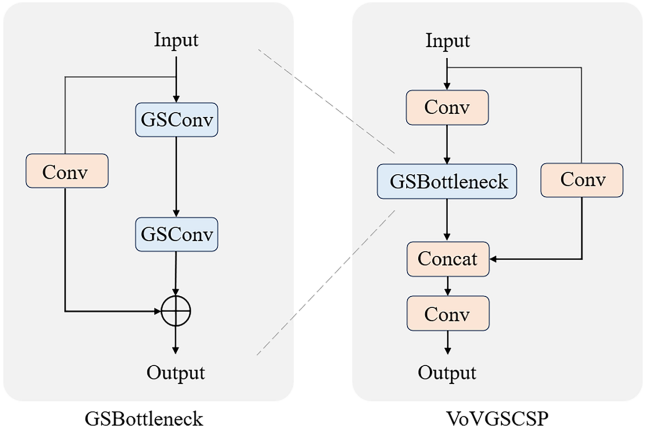


Figure 6: Schematic diagram of the VoVGSCSP structure.

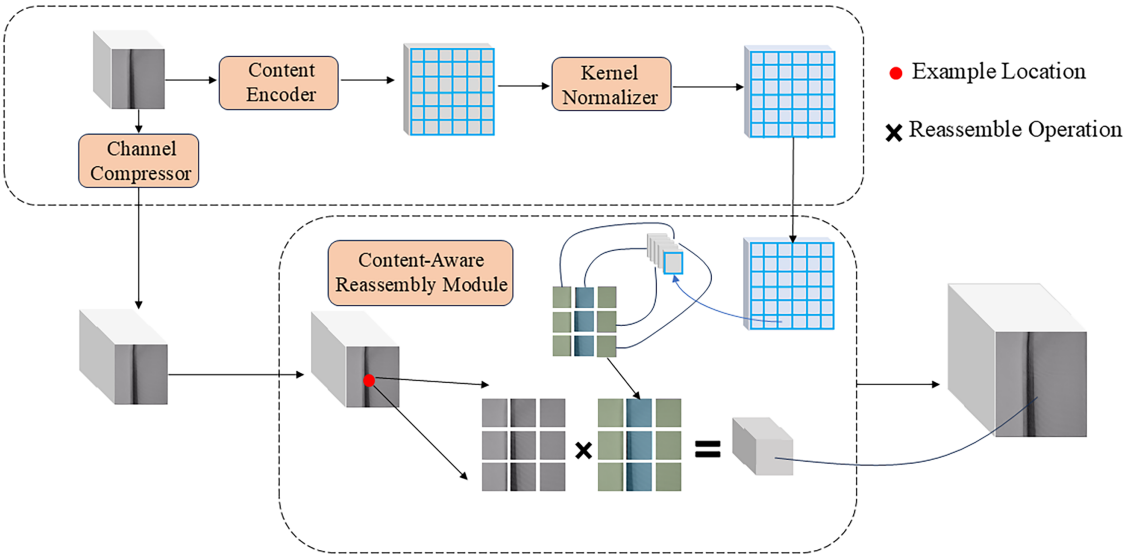


Figure 7: Schematic diagram of the CARAFE structure.

CARAFE comprises two core components:

Kernel Prediction Module: This module is designed to derive the reassembly kernels. Channel Compression: A 1 × 1 convolution is utilized to compress the number of input channels to Cm, reducing subsequent computational load and focusing the network on key feature information. Content Encoding: A convolutional layer encodes the compressed content features to generate an upsampling kernel prediction map with a shape of $\sigma H \times \sigma W \times k_{up}$. Pixel Shuffle: The predicted kernels are rearranged in the spatial dimension to establish a one-to-one correspondence with the spatial locations of the output feature map.

CARAFE: This module performs a weighted dot product between the predicted kernels and the pixels within the corresponding neighborhood of the input feature map. The output feature map at coordinate (i', j') is calculated as follows:

$$Y(i', j') = \sum_{n=-r}^{r} \sum_{m=-r}^{r} W_{i',j'}(n, m) \cdot X(i + n, j + m) \tag{3}$$

where $W_{i',j'}$ denotes the dynamically generated reassembly kernel of size $k_{up} \times k_{up}$, X represents the corresponding neighborhood in the input feature map, and r is the kernel radius.

By aligning the upsampling kernels with the local texture and shape features of the defects, CARAFE effectively preserves edge continuity and enhances the reconstruction quality of minute anomalous regions.

3.5 Fusion of the EMA Mechanism

The dense and repetitive hook-like structures in the background of hook-and-loop fasteners manifest as prominent high-frequency noise, which easily obscures the low-frequency signals of defects such as oil stains. To enhance the network's perceptual focusing capability on defect regions, the EMA module is introduced, as illustrated in Fig. 8.

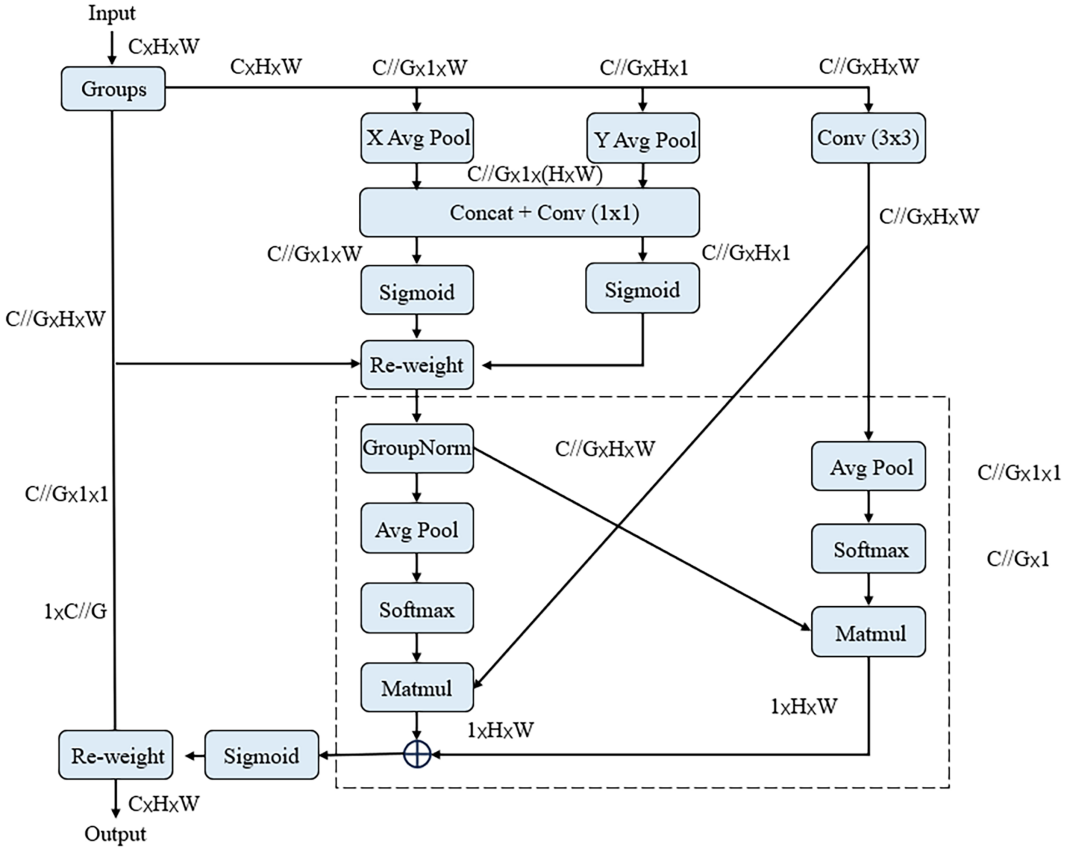


Figure 8: Schematic diagram of the EMA module structure.

EMA partitions the input feature map into G sub-groups and utilizes two parallel branches to extract features: a 1 × 1 convolution branch to capture pixel-level dependencies across channels, and a 3 × 3 convolution branch to capture local spatial contextual information. Subsequently, the cross-spatial learning module aggregates these features through 2D global average pooling and generates attention maps via an activation function. The final recalibrated output Y_{out} is obtained through element-wise addition as follows:

$$Attn = \sigma\left(F_{agg}\left(X_{1\times1}, X_{3\times3}\right)\right) \tag{4}$$

$$Y_{out} = X_{1\times1} \cdot Attn_1 + X_{3\times3} \cdot Attn_2 \tag{5}$$

where σ signifies the Sigmoid, and F_{agg} indicates the cross-spatial aggregation operator. EMA is integrated at the tail of the backbone to refine high-level semantic features, and at the end of the neck to recalibrate the fused multi-scale features, thereby significantly enhancing the localization precision of small-target defects.

3.6 Dynamic Non-Monotonic Focusing Loss Function

The Complete Intersection over Union (CIoU), employed by default in YOLOv8, tends to force the model to overfit low-quality annotated samples with ambiguous boundaries. This phenomenon generates harmful gradients and degrades the generalization capability of the model. To accelerate bounding box regression convergence, the WIoUv3 loss function is adopted in this paper.

WIoUv3 introduces an adaptive non-monotonic focusing strategy that evaluates sample quality by calculating the outlier degree β of anchor boxes and dynamically allocates gradient gain r. The mathematical expression for this loss function is:

$$L_{WIoUv3} = r \cdot L_{WIoUv1} \tag{6}$$

$$L_{WIoUv1} = \exp\left(\frac{\left(b_{c_x}^{gt} - b_{c_x}\right)^2 + \left(b_{c_y}^{gt} - b_{c_y}\right)^2}{\left(c_w^2 + c_h^2\right)^2}\right)(1 - IoU) \tag{7}$$

L_{WIoUv1} consists of two components. The first part is the exponential distance penalty term: $\exp\left(\frac{\left(b_{c_x}^{gt} - b_{c_x}\right)^2 + \left(b_{c_y}^{gt} - b_{c_y}\right)^2}{\left(c_w^2 + c_h^2\right)^2}\right)$. $\left(b_{c_x}^{gt} - b_{c_x}\right)^2 + \left(b_{c_y}^{gt} - b_{c_y}\right)^2$ denotes the central Euclidean distance between the predicted and ground-truth boxes. The term $\left(c_w^2 + c_h^2\right)^2$ typically denotes the squared diagonal length of the smallest enclosing bounding box. The second part is the basic IoU loss: $(1 - IoU)$. This serves as the foundation for all bounding box regression losses, reflecting the degree of overlap between the predicted box and the ground-truth box. A larger IoU yields a smaller loss; when the two boxes overlap completely, $1 - IoU = 0$.

Where r is the non-monotonic focusing coefficient. By dynamically assessing sample quality via $\beta = \frac{L_{IoU}}{L_{IoU}^*}$, The outlier degree β is utilized as a metric to evaluate the quality of the predicted bounding boxes, where L_{IoU}^* represents the moving average of the IoU loss across iterations to serve as a dynamic baseline. A smaller β indicates the spatial overlap between the prediction and the ground truth, categorizing them as high-quality samples. Conversely, a larger β means a bigger deviation, marking these anchors as low-quality samples or extreme outliers.

This mechanism gives lower weights to simple samples and extreme outliers. Instead, it focuses maximum gradient gain on medium-quality samples of moderate difficulty. This approach speeds up model convergence and handles annotation noise better in real industrial settings.

3.7 Overall Algorithmic Flow of GSL-YOLOv8-EMA

Algorithm 1 outlines the training process of GSL-YOLOv8-EMA, and the flowchart shows the overall workflow. The C2f-Faster backbone extracts initial features, which the Slim-neck then fuses across scales. Bounding box optimization relies on the WIoUv3 loss function. This design reduces background noise while keeping detection speed high.

Algorithm 1: The training pipeline of GSL-YOLOv8-EMA

Input: Training dataset $D = \{(X_i, Y_i)\}_{i=1}^{N}$; Total epochs E; Learning rate η.
Output: Optimal model weights W^*.
Initialize: Model weights W.
for $epoch = 1$ **to** E **do**
  for each mini-batch (X, Y) in D **do**
    Extract multi-scale features from X via **C2f-Faster** and purify the highest-level semantic features using **EMA**.
    Execute Top-Down and Bottom-Up feature fusion using **CARAFE**, **VoVGSCSP**, and **EMA**.
    Generate predictions (Pred_cls, Pred_box) via the Detection Head.
    Calculate outlier degree β and dynamically allocate gradient gain r.
    Compute total loss L_{total} weighted by r via **WIoUv3**.
    Update weights W via backpropagation based on L_{total} and η.
  end for
end for
return W^*

4 Experiment

To evaluate the performance of GSL-YOLOv8-EMA in hook-and-loop defect detection, this section presents four main parts. We first describe the experimental setup and evaluation metrics. Next, we conduct ablation studies to test each module. We then compare our model with mainstream algorithms and test its generalization on public datasets.

4.1 Experimental Setup and Performance Metrics

4.1.1 Experimental Platform and Hyperparameters

To keep conditions fair, all experiments used the same hardware and software setup. The models were implemented using the PyTorch framework, utilizing CUDA for GPU-accelerated inference. Detailed experimental platform configurations are provided in Table 1.

Table 1: Configuration of experimental hardware and software environment.

Configuration Item	Parameters
Operating System	Windows 11
GPU	NVIDIA GeForce RTX 4090
Deep Learning Framework	PyTorch 2.6.0, Python 3.10
Acceleration Libraries	CUDA 11.8, cuDNN 8.5

Considering the limited sample size of the custom-built hook-and-loop fastener dataset, the following training strategies summarized in Table 2 are adopted to improve model generalizability and mitigate overfitting:

To ensure the statistical reliability of the results, both the baseline YOLOv8n and the proposed GSL-YOLOv8-EMA were evaluated across 5 independent training runs using different random seeds. The metrics for our custom dataset are reported as Mean ± Standard Deviation (SD).

Table 2: Configuration of training hyperparameters.

Hyperparameter	Value
Optimizer	SGD
Base Learning Rate	0.01
Final Learning Rate	0.001
Momentum	0.937
Weight Decay	0.0005
Batch Size	8
Total Epochs	300
Early Stopping	50

4.1.2 Performance Metrics

This study selects the Confusion Matrix, Precision (P), Recall (R), Mean Average Precision (mAP), Parameters, and Frames Per Second (FPS) as the core evaluation metrics.

Precision and Recall are as follows:

$$P = \frac{TP}{TP + FP} \quad R = \frac{TP}{TP + FN} \tag{8}$$

where TP (True Positives) denotes correctly identified positive samples, FP (False Positives) represents falsely detected samples, and FN (False Negatives) refers to missed samples.

The definitions of Average Precision (AP) for a single category and mAP across all categories are as follows:

$$AP = \int_0^1 P(R)\, dR \tag{9}$$

$$mAP = \frac{1}{N} \sum_{i=1}^{N} \int_0^1 P_i(R_i)\, dR \tag{10}$$

This study focuses on the mAP at an Intersection over Union (IoU) threshold of 0.5, denoted as mAP@0.5.

In industrial defect detection, recall is the most important single metric. For hook-and-loop fasteners, missing a defect is much worse than a false alarm. This is because defective products must not enter production. Meanwhile, mAP@0.5 checks whether the model maintains high recall without dropping precision too much.

4.2 Augmentation Strategy

The performance comparison between the models with and without data augmentation is summarized in Table 3. The implementation of data augmentation brings a clear performance growth. Meanwhile, as illustrated in Fig. 9, the loss curves confirm that the augmented model avoids overfitting. This training strategy ensures high precision and accuracy simultaneously.

The loss curve without data augmentation shows big fluctuations, which indicates serious overfitting. In contrast, the loss curve with data augmentation declines smoothly showing better results.

Table 3: Comparison of model performance with and without augmentation.

Model	Augmentation	mAP@0.5 (%)	Precision	Recall	F1
GSL-YOLOv8-EMA	No	68.4	0.65	0.61	0.63
	Yes	88.2	0.87	0.89	0.88

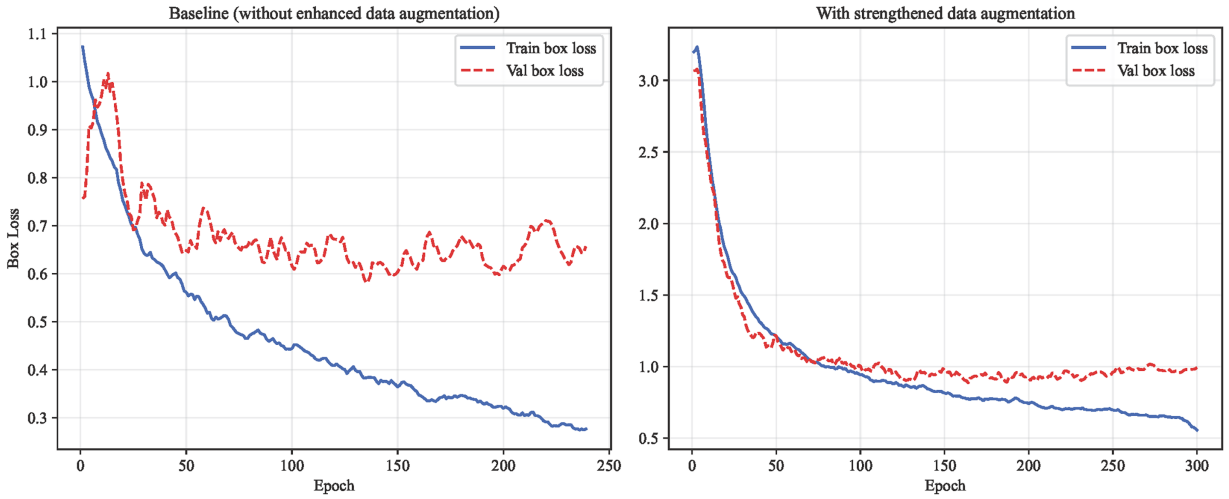


Figure 9: Comparison of the loss curves between the original and augmented datasets.

4.3 Ablation Study of Improved Modules

To quantify the marginal contributions of the PConv, GSConv, CARAFE, EMA, and Wise-IoU modules to the overall model performance, this paper designs a sequence of incremental ablation experiments using YOLOv8n as the starting point.

4.3.1 Quantitative Results Analysis

The objective metric evaluation results under different module combinations are shown in Table 4.

Table 4: Ablation study of GSL-YOLOv8-EMA model.

Group	YOLOv8n	PConv	Slim-Neck	CARAFE	EMA	WIOU	mAP@0.5 (%)	Params (M)	FLOP (G)
A	√						78.4	3.1	8.2
B	√	√					74.2	2.2	6.2
C	√	√	√				74.8	2.1	6.3
D	√	√	√	√			77.3	2.2	6.4
E	√	√	√	√	√		81.4	2.3	6.5
F	√	√	√	√	√	√	88.2	2.3	6.4

The evolution of the experimental data can be divided into three key stages:

1. Lightweight Reconstruction Stage (Groups B and C): With the incorporation of the FasterNet/PConv backbone and the Slim-neck architecture, the model's parameter count was significantly compressed to

2.1 M, and the computational complexity was substantially reduced. Although the mAP@0.5 dropped to 74.8% during this phase, this is considered a typical trade-off phenomenon in lightweight processing.

2. Feature Enhancement Stage (Groups D and E): The introduction of the CARAFE operator leads to a rapid recovery of the mAP@0.5 to 77.3%. This modification reduces feature distortion caused by traditional interpolation. As a result, the model performs much better on slender "Stained stripe" defects. Adding the EMA mechanism further raises mAP@0.5 to 81.4%, beating the baseline and reducing background noise.
3. Loss Optimization Stage (Group F): After using the WIoU loss function, the model's performance reached its best. Compared to the baseline, GSL-YOLOv8-EMA increased mAP@0.5 by 9.8% to 88.2% while keeping the number of parameters low. These results confirm the dynamic focusing mechanism improves feature learning on low-quality samples.

4.3.2 Qualitative Analysis and Visualization

As shown in Fig. 10, the prediction accuracy is relatively high, with 'stained-stripe' at 91% and 'stain' at 89%. Compared with the baseline model, the P-R curve of this network is fuller. These metrics indicate that the improved model performs well.

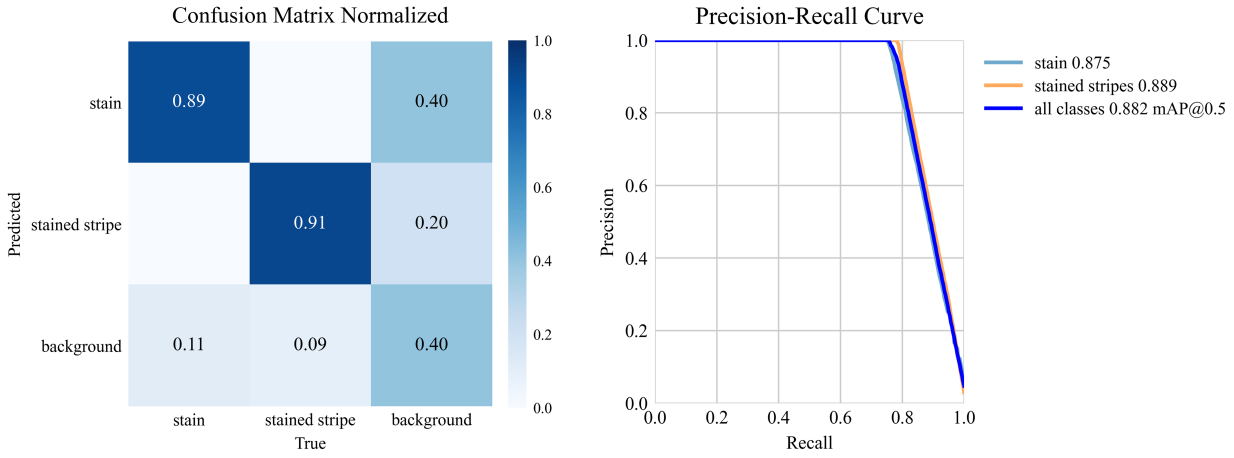


Figure 10: Confusion matrix and P-R curves of the GSL-YOLOv8-EMA model. (**a**) Normalized confusion matrix of the two defect classes and the background, where rows denote predicted classes and columns denote true classes; (**b**) Precision–recall curves with AP of 0.875 (stain), 0.889 (stained stripes), and mAP@0.5 of 0.882 (all classes).

4.4 Comparative Analysis

To verify the effectiveness of this algorithm in textile defect detection, we conducted a comparative evaluation of models of the same type on our dataset. For a fair comparison, all models were trained from scratch under the same parameters, as summarized in Table 5.

GSL-YOLOv8-EMA achieved 88.2 ± 3.1% mAP@0.5, surpassing the baseline model. Its small fluctuations reflect that the model possesses strong stability. Comparison experiments with SOTA models reveal a limitation: general models often perform poorly on industrial textures with strong backgrounds and weak features. Our architecture successfully achieves high-precision detection with a lightweight design.

Table 5: Comparison of performance across different algorithms.

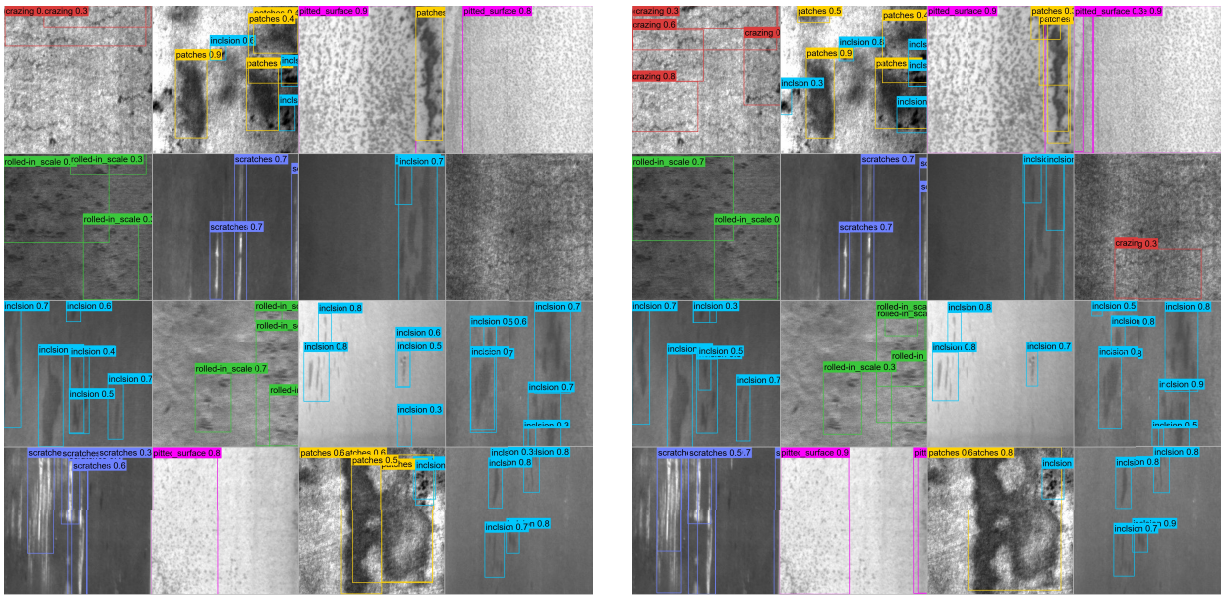
Model	Type	mAP@0.5 (%)	Precision	Recall	F1	Params (M)	FPS	FLOPs (G)
Faster R-CNN	ResNet-50	66.4 ± 7.2	0.72 ± 0.06	0.66 ± 0.08	0.70 ± 0.06	41.5	22	182
RT-DETR	ResNet-50	69.0 ± 7.0	0.74 ± 0.05	0.67 ± 0.07	0.71 ± 0.08	42.0	61	125
YOLOv9t	GELAN-net	73.1 ± 4.2	0.76 ± 0.05	0.78 ± 0.04	0.77 ± 0.06	2.0	118	7.8
YOLOv10n	SCDown-Net	67.6 ± 6.1	0.62 ± 0.05	0.61 ± 0.06	0.61 ± 0.08	2.3	130	8.6
YOLO11n	C3k2-Net	74.8 ± 4.4	0.73 ± 0.03	0.78 ± 0.02	0.76 ± 0.02	2.5	136	6.4
YOLO12n	A2-Net	70.2 ± 5.8	0.70 ± 0.04	0.68 ± 0.04	0.69 ± 0.05	2.6	134	6.5
YOLOv8n	CSPDark-net	78.4 ± 5.2	0.78 ± 0.03	0.74 ± 0.04	0.76 ± 0.03	3.0	125	8.3
GSL-YOLOv8-EMA	Faster-net	88.2 ± 3.1	0.87 ± 0.02	0.89 ± 0.03	0.88 ± 0.03	2.2	148	6.4

4.5 Generalization Capability Verification

To evaluate the generalization performance of GSL-YOLOv8-EMA across various industrial texture defect detection tasks, we extended our evaluation to two widely recognized public benchmarks: NEU-DET (steel surface defects) and TILDA (textile textures).

4.5.1 Generalization Validation on Metal Textures Based on NEU-DET

Industrial inspection relies on studying metal surface defects [24]. The NEU-DET dataset exhibits complex background textures and low contrast. These are very similar to our dataset, as shown in Fig. 11.



(**a**) YOLOv8n (**b**) GSL-YOLOv8n-EMA

Figure 11: Comparison of detection results on the NEU-DET dataset.

4.5.2 Comparative Analysis of the NEU-DET Dataset

We benchmark our method on the NEU-DET. The results is shown in Table 6.

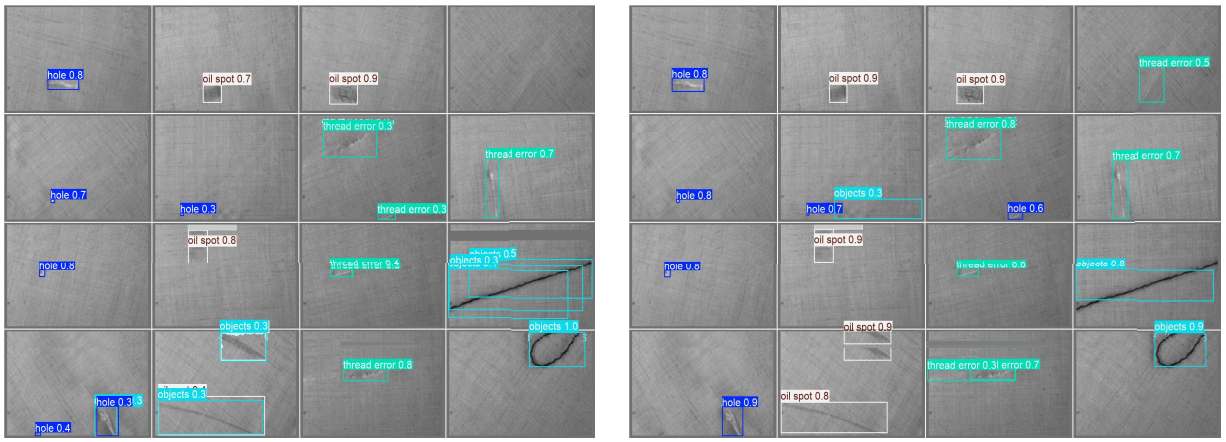
Table 6: Comparison of performance across different algorithms on the NEU-DET dataset.

Model	Type	mAP@0.5 (%)	Precision	Recall	F1	Params (M)	FPS	FLOPs (G)
YOLOv9t	GELAN-net	80.2	0.80	0.78	0.79	2.0	118	7.8
YOLOv10n	SCDown-Net	76.0	0.72	0.73	0.72	2.3	130	8.6
YOLO11n	C3k2-Net	78.9	0.78	0.76	0.77	2.5	136	6.4
YOLO12n	A2-Net	78.1	0.75	0.74	0.75	2.6	134	6.5
YOLOv8n	CSPDark-net	76.2	0.76	0.75	0.75	3.0	128	8.3
GSL-YOLOv8-EMA	Faster-net	80.8	0.81	0.78	0.80	2.2	148	6.4

This qualitative analysis shows that the improved model adapts well to complex metal textures like scale and water stains. Specifically, the model reduces the miss rate for defects like cracks and scratches. This proves it has strong background noise suppression.

4.5.3 Generalization Validation on Textile Textures Based on TILDA

The TILDA dataset includes repeating fabric textures and random background noise. Factors like lighting changes and viewpoint switches cause this kind of noise. These changes pose challenges for feature extraction and designing for interference resistance, as shown in Fig. 12.



(**a**) YOLOv8n (**b**) GSL-YOLOv8n-EMA

Figure 12: Comparison of detection results on the TILDA dataset.

4.5.4 Comparative Analysis of the TILDA Dataset

We benchmark our method on the TILDA. The results are shown in Table 7.

The visualization results indicate that the model can effectively extract tiny defects from repetitive grid textures. And it will reduce the false positive and false negative rates. Cross-dataset validation demonstrates the network's adaptability to scenarios with strong textures and weak features.

Table 7: Comparison of performance across different algorithms on the TILDA dataset.

Model	Type	mAP@0.5 (%)	Precision	Recall	F1	Params (M)	FPS	FLOPs (G)
YOLOv9t	GELAN-net	86.2	0.88	0.76	0.82	2.0	118	7.8
YOLOv10n	SCDown-Net	89.8	0.83	0.89	0.86	2.3	130	8.6
YOLO11n	C3k2-Net	87.3	0.90	0.87	0.89	2.5	136	6.3
YOLO12n	A2-Net	89.7	0.89	0.91	0.90	2.6	134	6.5
YOLOv8n	CSPDark-net	88.4	0.90	0.88	0.89	3.0	128	8.2
GSL-YOLOv8-EMA	Faster-net	91.8	0.91	0.93	0.92	2.2	148	6.3

5 Discussion

Our GSL-YOLOv8-EMA network achieves a practical balance in the detection of hook-and-loop fasteners. Compared to classic two-stage architectures such as Faster R-CNN [25,26], our model not only improves accuracy but also boosts inference speed from 22 FPS to 148 FPS. When compared with recent YOLO variants, it avoids the common pitfall of trading off latency for precision. For example, relative to GSL-YOLOv8n [27], our model achieves a 9.8% increase in mAP@0.5 over the baseline. In comparison with YOLOv8n-LAW [28] and YOLO-BGS [29], our model maintains higher accuracy gains while still delivering a high frame rate of 148 FPS. In contrast to SRSE-YOLO [30], our design also incorporates a high-resolution detection head.

These findings apply to industrial scenarios with high noise and repetitive textures. They suggest that feature reconstruction and attention mechanisms are more important than the size of the parameters. By reducing the initial weights through PConv, the mAP dropped from 78.4% to 74.8%. It help us avoid parameter expansion. CARAFE and EMA optimize resource allocation. With only 6.4G FLOPs, the network can achieve an mAP of 88.2 ± 3.1% at 148 FPS.

Among all integrated modules, WIoUv3 made the most significant contribution to performance improvement. It increases mAP@0.5 by 6.8% without incurring any additional computational cost. Regarding module collaboration, PConv and Slim-neck first freed up computational resources by reducing computational redundancy. CARAFE and EMA utilized these resources to recover subtle details and suppress background noise. WIoUv3 further optimized the model's ability to localize complex samples. This clear module synergy enabled this framework to successfully overcome the trade-off between speed and accuracy.

Several practical constraints remain. While data augmentation enhanced the dataset size, it still pales next to massive benchmarks like ImageNet [31]. The current setup focuses strictly on superficial flaws overlooking structural damage such as broken hooks. On the environmental side, controlled lab lighting fails to mimic real-world challenges like motion blur or severe illumination shifts. Further, high-end GPUs are indispensable for current tests. These limitations point toward several clear paths for future work. To counter data scarcity, we aim to synthesize high-quality virtual samples via generative adversarial networks. Translating these gains into real-world industrial deployment will involve hardware optimization on edge devices using inference engines like TensorRT.

6 Conclusion

This paper proposes a lightweight algorithm named GSL-YOLOv8-EMA, which detects small defects in hook-and-loop fasteners in real time despite strong background noise. We constructed a custom dataset

using a dedicated vision platform and applied combined augmentation strategies to prevent overfitting. Algorithmically, we integrated FasterNet, PConv, and GSConv to reduce computational overhead. Furthermore, CARAFE upsampling and EMA attention were introduced to enhance weak defect features while suppressing noise, and Wise-IoU was used to optimize bounding box regression. Experiments show that our model achieves an mAP@0.5 of 88.2 ± 3.1% at 148 FPS, outperforming the baseline by 9.8%, and generalizes well on the NEU-DET and TILDA datasets.

This research not only successfully balances detection accuracy and inference speed, but also provides an efficient and lightweight new method for industrial micro target detection under complex texture backgrounds, which has important theoretical research value and engineering application prospect.

Acknowledgement: We thank Prof. Xiuxiang Huang for the support in the experiment platform establishment. During the manuscript writing and revision, we adopted AI-assisted language editing to improve the quality of the manuscript.

Funding Statement: This work was supported by Hunan Provincial Social Science Fund Project titled "Research on Huxiang Brocade Pattern Generation and Human–Machine Collaboration Mechanisms Based on LoRA Models" (Grant No: 23A0044).

Author Contributions: Jiaji Pan: Conception, Resources. Yanxin Xiang: Investigation, Methodology, Writing-original draft. Fujun Liu: Data curation. Wei Wang: Investigation, Supervision. All authors reviewed and approved the final version of the manuscript.

Availability of Data and Materials: Data will be available upon request.

Ethics Approval: Not applicable.

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

References

1. Jiao J, Zhang F, Jiao T, Gu Z, Wang S. Bioinspired superdurable pestle-loop mechanical interlocker with tunable peeling force, strong shear adhesion, and low noise. Adv Sci. 2018;5(4):1700787. doi:10.1002/advs.201700787.
2. Kumar A. Neural network based detection of local textile defects. Pattern Recognit. 2003;36(7):1645–59. doi:10.1016/S0031-3203(03)00005-0.
3. Ameri R, Hsu CC, Band SS. A systematic review of deep learning approaches for surface defect detection in industrial applications. Eng Appl Artif Intell. 2024;130(3):107717. doi:10.1016/j.engappai.2023.107717.
4. Chen T, Kornblith S, Norouzi M, Hinton G. A simple framework for contrastive learning of visual representations. In: Proceedings of the International Conference on Machine Learning; 2020 Jul 13; Online. New York, NY, USA: The Association for Computing Machinery (ACM); 2020. p. 1597–607.
5. Bodnarova A, Bennamoun M, Latham S. Optimal Gabor filters for textile flaw detection. Pattern Recognit. 2002;35(12):2973–91. doi:10.1016/S0031-3203(02)00017-1.
6. Dunn D, Higgins WE. Optimal Gabor filters for texture segmentation. IEEE Trans Image Process. 1995;4(7):947–64. doi:10.1109/83.392336.
7. Ojala T, Pietikäinen M, Harwood D. A comparative study of texture measures with classification based on featured distributions. Pattern Recognit. 1996;29(1):51–9. doi:10.1016/0031-3203(95)00067-4.
8. Ojala T, Pietikainen M, Maenpaa T. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. IEEE Trans Pattern Anal Mach Intell. 2002;24(7):971–87. doi:10.1109/TPAMI.2002.1017623.
9. Haralick RM. Statistical and structural approaches to texture. Proc IEEE. 1979;67(5):786–804. doi:10.1109/proc.1979.11328.
10. Haralick RM, Shanmugam K, Dinstein I. Textural features for image classification. IEEE Trans Syst. 1973; SMC-3(6):610–21. doi:10.1109/tsmc.1973.4309314.
11. Cortes C, Vapnik V. Support-vector networks. Mach Learn. 1995;20(3):273–97. doi:10.1023/A:1022627411411.

12. Chen J, Mei S, Ren Z, Tang L, Xu B, Fu G, et al. SDLS-YOLO: an ultra-lightweight real-time fabric defect detection algorithm based on multi-module coupling. J King Saud Univ Comput Inf Sci. 2025;38(1):7. doi:10.1007/s44443-025-00376-w.
13. Li J, Kang X. Mobile-YOLO: an accurate and efficient three-stage cascaded network for online fiberglass fabric defect detection. Eng Appl Artif Intell. 2024;134(20):108690. doi:10.1016/j.engappai.2024.108690.
14. Chen J, Kao SH, He H, Zhuo W, Wen S, Lee CH, et al. Run, don't walk: chasing higher FLOPS for faster neural networks. In: Proceedings of the 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR); 2023 Jun 17–24; Vancouver, BC, Canada. New York NY, USA: IEEE; 2023. p. 12021–31. doi:10.1109/CVPR52729.2023.01157.
15. Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, et al. Attention is all you need. In: Proceedings of the 31st Conference on Neural Information Processing System (NIPS 2017); 2017 Dec 4; Long Beach, CA, USA. New York, NY, USA: The Association for Computing Machinery (ACM); 2017. p. 6000–10.
16. Dong H, Song K, He Y, Xu J, Yan Y, Meng Q. PGA-Net: pyramid feature fusion and global context attention network for automated surface defect detection. IEEE Trans Ind Inform. 2020;16(12):7448–58. doi:10.1109/TII.2019.2958826.
17. Zhao Y, Liu Q, Su H, Zhang J, Ma H, Zou W, et al. Attention-based multiscale feature fusion for efficient surface defect detection. IEEE Trans Instrum Meas. 2024;73:1–10. doi:10.1109/tim.2024.3372229.
18. Li Y, Zhao W, Pan J. Deformable patterned fabric defect detection with fisher criterion-based deep learning. IEEE Trans Automat Sci Eng. 2017;14(2):1256–64. doi:10.1109/tase.2016.2520955.
19. Wang J, Chen K, Xu R, Liu Z, Loy CC, Lin D. CARAFE: content-aware ReAssembly of FEatures. In: Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision (ICCV); 2019 Oct 27–Nov 2; Seoul, Republic of Korea. p. 3007–16. doi:10.1109/iccv.2019.00310.
20. Wang J, Chen K, Xu R, Liu Z, Loy CC, Lin D. CARAFE++: unified content-aware ReAssembly of FEatures. IEEE Trans Pattern Anal Mach Intell. 2021;44(9):4674–87. doi:10.1109/tpami.2021.3074370.
21. Tong Z, Chen Y, Xu Z, Yu R. Wise-IoU: bounding box regression loss with dynamic focusing mechanism. arXiv:230110051. 2023.
22. Ouyang D, He S, Zhang G, Luo M, Guo H, Zhan J, et al. Efficient multi-scale attention module with cross-spatial learning. In: Proceedings of the ICASSP 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP); 2023 Jun 4–10; Rhodes Island, Greece. p. 1–5. doi:10.1109/ICASSP49357.2023.10096516.
23. Bochkovskiy A, Wang CY, Liao HYM. YOLOv4: optimal speed and accuracy of object detection. arXiv:200410934. 2020.
24. Usamentiaga R, Lema DG, Pedrayes OD, Garcia DF. Automated surface defect detection in metals: a comparative review of object detection and semantic segmentation using deep learning. IEEE Trans Ind Appl. 2022;58(3):4203–13. doi:10.1109/TIA.2022.3151560.
25. Ren S, He K, Girshick R, Sun J. Faster R-CNN: towards real-time object detection with region proposal networks. IEEE Trans Pattern Anal Mach Intell. 2017;39(6):1137–49. doi:10.1109/TPAMI.2016.2577031.
26. Chen M, Yu L, Zhi C, Sun R, Zhu S, Gao Z, et al. Improved faster R-CNN for fabric defect detection based on Gabor filter with Genetic Algorithm optimization. Comput Ind. 2022;134:103551. doi:10.1016/j.compind.2021.103551.
27. Ma S, Liu Y, Zhang Y. Fabric defect detection based on improved lightweight YOLOv8n. Appl Sci. 2024;14(17):8000. doi:10.3390/app14178000.
28. Mei S, Shi Y, Gao H, Tang L. Research on fabric defect detection algorithm based on improved YOLOv8n algorithm. Electronics. 2024;13(11):2009. doi:10.3390/electronics13112009.
29. Lu G, Xiong T, Wu G. YOLO-BGS optimizes textile production processes: enhancing YOLOv8n with bi-directional feature pyramid network and global and shuffle attention mechanisms for efficient fabric defect detection. Sustainability. 2024;16(18):7922. doi:10.3390/su16187922.
30. Wang T, Dai N, Hu X, Xu K, Yuan Y. SRSE-YOLO: an improved fabric defect detection method for YOLOv8. Text Res J. 2025;16:11. doi:10.1177/00405175251397605.
31. Krizhevsky A, Sutskever I, Hinton GE. ImageNet classification with deep convolutional neural networks. Adv Neural Inf Process Syst. 2012;25(6):1–9. doi:10.1145/3065386.