applied sciences

MDPI

Article

Enhanced YOLOv8s with Multi-Teacher Distillation for Steel Cord Ply Defect Detection

Peng Huang [1], Zhongyi Xie [1], Rui Long [1], Feiqiang Zhou [1], Xinlong Zhang [1], Zejie Ke [1] and Guangzhan Huang [2,*]

[1] Key Laboratory of Advanced Manufacturing and Automation Technology for Guangxi Universities, College of Mechanical and Control Engineering, Guilin University of Technology, Guilin 541006, China; huangpeng@glut.edu.cn (P.H.); xiezhongyi008@outlook.com (Z.X.); longlongrui18@gmail.com (R.L.); zfq112517@gmail.com (F.Z.); zxi804823655@gmail.com (X.Z.); 1376331031a@gmail.com (Z.K.)

[2] Guilin Zhonghao Lichuang Mechanical & Electrical Equipment Company, Guilin 541004, China

* Correspondence: huangguangzhan008@163.com; Tel.: +86-182-0773-1750

Abstract

To improve detection accuracy for color-sensitive and small-target defects in steel cord ply, this paper introduces an improved YOLOv8s algorithm using multi-teacher stepwise hierarchical knowledge distillation for better adaptation across production lines. The improvements include: replacing the initial backbone convolutional layer with RGBV grouped convolution to enhance color feature extraction; substituting the SPPF module with SPPFCSPC-LSKA to improve multi-scale perception; and optimizing bounding box accuracy with the WIoU loss function. The multi-teacher distillation approach first transfers color feature learning using an RGBV-only teacher, then multi-scale feature learning with an SPPFCSPC-LSKA-only teacher. Experimental results show the improved model achieved 90.4% precision, 92.0% recall, 91.2% F1-score, and 97.2% mAP@0.5, surpassing the baseline YOLOv8s by 1.9, 2.2, 2.1, and 3.4 percentage points, respectively. The proposed model also achieves an inference time of 3.9 ms, representing a 1.0 ms reduction compared to the baseline. On a smaller dataset from another production line, single-teacher distillation increased precision, recall, F1-score, and mAP@0.5 to 84.6%, 82.0%, 83.3%, and 88.8%, respectively, albeit with an increase in inference time. The multi-teacher strategy further increased metrics to 97.5% precision, 88.8% recall, 92.9% F1-score, and 94.3% mAP@0.5, providing additional gains over single-teacher distillation while maintaining the same parameter count of 11.127 M and achieving a faster inference time of 4.1 ms on the target production line.

Keywords: steel cord ply; defect detection; YOLOv8s; RGBV grouped convolution; SPPFCSPC-LSKA; WIoU loss; knowledge distillation; multi-teacher

Check for updates

Received: 3 February 2026
Revised: 28 March 2026
Accepted: 31 March 2026
Published: 13 April 2026

Copyright: © 2026 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license.

1. Introduction

Automotive safety is a top priority in transportation engineering. Tires, as the only components in contact with the road, play a crucial role in driving stability and occupant safety. Steel cord ply forms the foundation of tire construction and is a key component of the belt layer [1]. During manufacturing, it is prone to surface defects such as localized cord exposure, splice splits, and wax marker traces. Undetected, these defects create uneven stress and weaken the tire structure, compromising safety. Wax marker traces and small defects, such as yellowish specks measuring 16 × 16 to 32 × 32 pixels in 16-megapixel images, are difficult to detect due to their size and reliance on color information. As the

industry moves from sampling-based to full inspection, steel cord ply defect detection increasingly relies on deep learning-based machine vision systems.

Deep learning-based object detection algorithms are divided into two main groups based on their processing pipelines. Two-stage algorithms include region-based Convolutional Neural Networks (R-CNN) [2], while single-stage algorithms include the Single Shot MultiBox Detector (SSD) [3] and the You Only Look Once (YOLO) series [4–7]. Single-stage methods generally have simpler network structures and provide faster detection. In recent years, deep learning has been extensively explored for a wide range of industrial and localization applications. For instance, Convolutional Neural Networks (CNNs) have been successfully applied to tasks such as defect detection, image classification, and indoor localization. In particular, the use of Generative Adversarial Networks (GANs) and semi-supervised learning has been investigated to address the challenge of limited labeled data, as demonstrated by Njima et al. [8] in the context of indoor localization, where artificially generated samples were shown to effectively supplement real measurements. For tire defect inspection, Saleh et al. [9] developed a hybrid deep learning model with an attention mechanism that improved both accuracy and robustness by combining global and local features. While these studies highlight deep learning's potential for tire defect detection, current methods still lack sufficient recognition accuracy for steel cord ply, particularly for color-sensitive and small-target defects.

Detecting color-sensitive defects on steel cord ply is challenging because these defects often lack distinct textures or shapes, making color feature perception essential. Traditional image processing methods address this by converting images into alternative color spaces. For example, Huang et al. [10] used the HSV color space and extracted hue, saturation, and value features to detect chromatic differences in motor control boards. However, reliance on preset thresholds limits their effectiveness for steel cord ply defect detection. These methods serve only as preprocessing and do not allow models to optimize color features autonomously.

Detecting small-target defects in steel cord ply remains a significant challenge. As network depth increases, gradient information for small-target defects often vanishes, leading to high false negative rates during deployment. Lu et al. [11] improved infrared small-target detection in complex backgrounds using a nested feature pyramid network with a disturbance-suppression loss function. Spatial Pyramid Pooling (SPP) [12] in YOLO-series algorithms addresses the fixed input dimension constraint of CNNs. The YOLOv5 authors introduced Spatial Pyramid Pooling Fast (SPPF), which accelerates inference speed but offers limited receptive field expansion. Spatial Pyramid Pooling Cross-Stage Partial Connections (SPPCSPC) [13], proposed in YOLOv7, uses a dual-path architecture for cross-stage feature fusion but increases inference latency due to more parameters. SPPFCSPC [14–16] combines the efficiency of SPPF with the fusion capacity of SPPCSPC. Heng et al. [17] applied SPPFCSPC in an assistive navigation system for visually impaired individuals, improving small-obstacle detection. Shi et al. [18] enhanced small-defect recognition in SiC crystal inspection using the SCDD-Net model with SPPFCSPC. While these methods improve small-target sensitivity, they still struggle to capture long-range contextual dependencies. Attention mechanisms have become essential for effective multi-scale feature aggregation. Large Separable Kernel Attention (LSKA) [19–21] has shown strong performance in this area. Liu et al. [22] added an SPPF-LSKA module to an enhanced YOLOv11 framework, reducing feature dilution in drone imagery through multi-scale fusion and global context modeling. Zhang et al. [23] developed a lightweight YOLOv8s-based model with SPPF-LSKA to improve detail preservation and interference resistance, increasing small-pedestrian detection accuracy in complex scenes. Despite these advances, high false negative rates persist in steel cord ply small-target defect detection.

Steel cord ply defect detection systems face two main challenges in industrial deployment, especially on new production lines. First, limited historical data reduces model performance in few-shot learning scenarios. Second, algorithms that work well on one production line often perform poorly when transferred to others or to different product specifications, due to deep learning's reliance on training data and limited transferability. To address the data scarcity challenge in industrial deployment, alternative data acquisition and modeling strategies have been explored in related fields. For instance, a 3D laser profilometry-based method for automatic defect detection in aluminum welds was proposed by Cardellicchio et al. [24], where high accuracy was achieved by extracting dense geometric features from a limited number of physical samples. In their work, it is demonstrated that real-time quality control under data-constrained conditions can be enabled by combining efficient data acquisition with lightweight machine learning models. This insight is leveraged to motivate the use of knowledge distillation, by which utility from small datasets in cross-production-line scenarios can be maximized. Knowledge Distillation (KD) is a proven approach for model compression and transfer learning. By transferring knowledge from pre-trained teacher models to compact student models, knowledge distillation improves inference efficiency and enhances model adaptability and generalization under data scarcity. He et al. [25] introduced a lightweight network with knowledge distillation, using a task-decoupling strategy to increase robustness for underwater seafood detection in few-shot settings, while reducing parameter counts and maintaining accuracy. Xu et al. [26] developed a crack segmentation model that combines pruning and channel-wise knowledge distillation to balance precision and efficiency in concrete crack quantification. Although these studies demonstrate KD's potential in data-scarce scenarios, the complex defect types in steel cord ply and variations between production lines require specialized adaptation. Developing customized knowledge distillation frameworks for this domain remains an open research problem that needs systematic investigation.

In steel cord ply defect detection, conflicts between optimization objectives of defect-specific enhancement modules can cause performance instability or degradation in integrated models. This also reduces the effectiveness of knowledge distillation when suboptimal models are used as teachers. Recently, multi-teacher knowledge distillation strategies have shown promise by leveraging the complementary strengths of structurally diverse or objective-divergent teacher models to improve a student's representational capacity. Yousefimehr et al. [27] proposed a multi-teacher knowledge distillation framework for anomaly detection, training multiple teachers on resampled datasets and distilling their knowledge into a lightweight student, which also enables model compression for efficient deployment. However, for steel cord ply defect detection, especially in cross-line and few-shot scenarios with multiple defect types, designing multi-teacher distillation strategies that resolve optimization conflicts and enable effective knowledge fusion remains an open challenge.

To address ongoing challenges in steel cord ply defect detection, including low accuracy for color-sensitive defects, high false negative rates for small-target defects, and limited cross-line transferability, this study focuses on the object detection task, where the model outputs both defect categories and their corresponding bounding box locations. Performance is evaluated using standard detection metrics, including precision, recall, Mean Average Precision (mAP), and F1-score.

The main contributions of this work are as follows.

- A RGBV grouped convolution module is proposed to enhance chromatic feature extraction by separating R, G, B, and V components at the network input stage;
- The SPPFCSPC-LSKA module is introduced to expand the receptive field and improve multi-scale feature fusion for small-target defect detection;

- Knowledge distillation is introduced to enable rapid cross-production-line adaptation under few-shot conditions, transferring knowledge from a high-accuracy teacher model to a compact student model with limited target-domain samples;
- A multi-teacher stepwise hierarchical distillation strategy is developed to integrate complementary strengths from structurally diverse teacher models, guiding the student through staged knowledge transfer while resolving optimization conflicts inherent in multi-defect scenarios.

2. Steel Cord Ply Defect Detection System

2.1. Detection Principle and System Architecture

The steel cord ply defect detection system integrates hardware and software to support real-time, high-precision surface defect detection. It is deployed on multiple production lines at Rubber Company A and Tire Company B, as shown in Figure 1. The main hardware components, including a line scan camera, tunnel light, and rotary encoder, are designed for continuous production at 50 m/min. The tunnel light, equipped with diffuser materials, provides uniform illumination to reduce ambient light interference and highlight defect features. The line scan camera, synchronized with the tunnel light, scans the moving fabric line by line. These scans are stitched together to create a complete image, preventing motion blur and gaps common with area scan cameras. The rotary encoder outputs pulse signals proportional to conveyor belt movement, compensating for speed fluctuations and preventing image distortion. Together, these components deliver a stable and reliable visual front-end for defect detection.

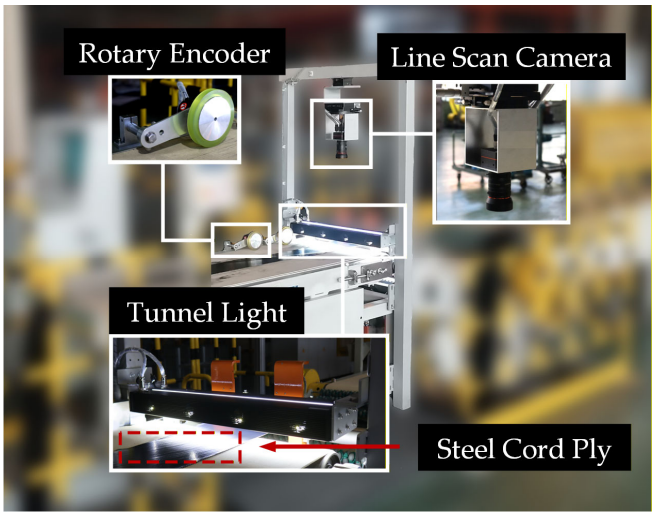


Figure 1. On-site deployment of the defect detection system.

The software platform acts as the central control hub, offering an intuitive interface and efficient real-time processing. As shown in Figure 2, the main interface uses a dual-pane display to monitor both the top and bottom surfaces of the steel cord ply. When a defect is detected, the system immediately displays annotated results. Control buttons at the top of the interface allow users to start or stop detection tasks for each surface independently. This platform supports a closed-loop workflow that includes image acquisition, defect recognition, and human interaction, which enhances automation and operational responsiveness.

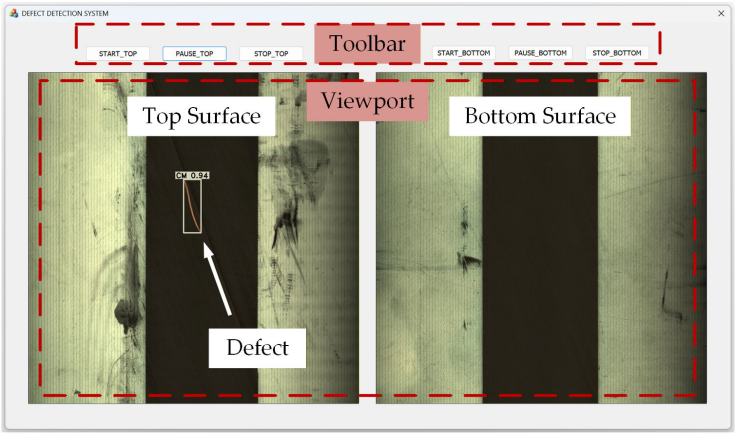


Figure 2. Software interface of the defect detection system.

2.2. Operational Workflow

The operational workflow is illustrated in Figure 3.

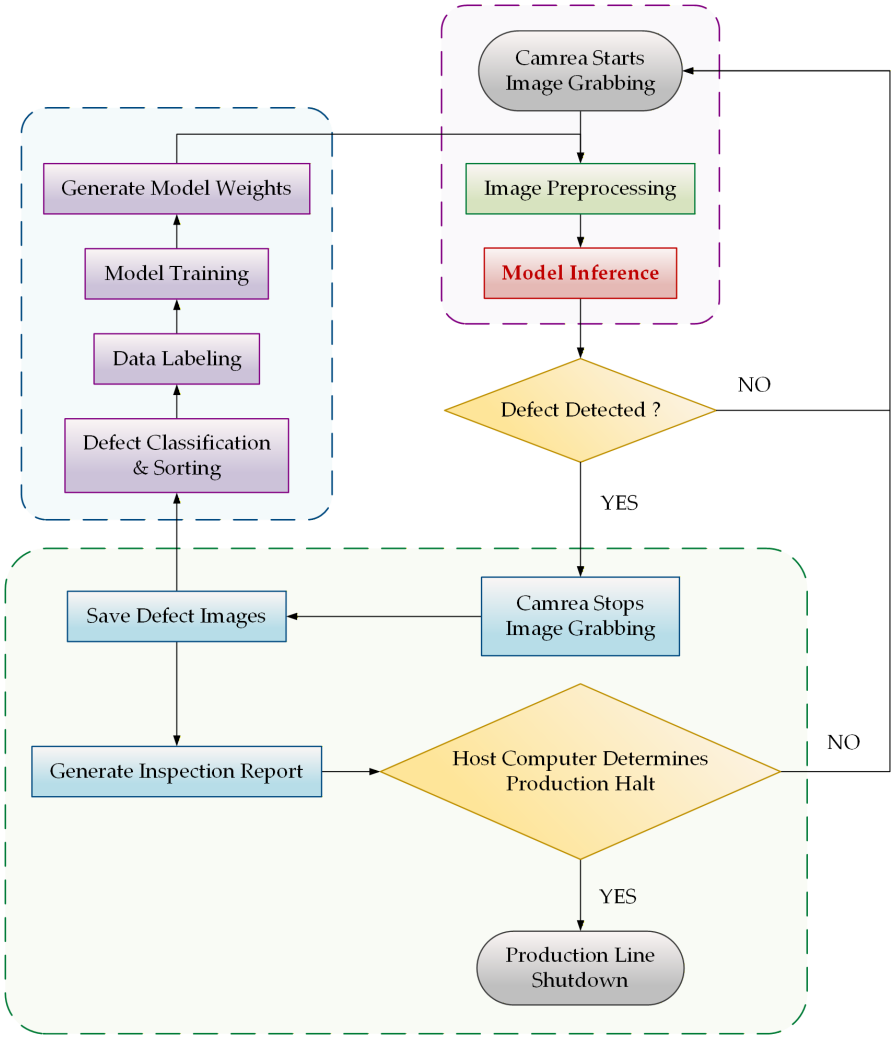


Figure 3. Workflow of the defect detection system. ("?": decision node.)

After initialization, the line-scan camera captures high-resolution, defect-free images of the steel cord ply. These images are preprocessed for dimensional standardization before analysis. The core processing stage uses a deep learning network to classify background and defect features based on trained weights. If recognition confidence exceeds a set

threshold, the system locates and categorizes defects, annotates their coordinates, and stores results in the database. The system also supports continuous improvement: new defect samples are manually verified and annotated, then added to the dataset for model retraining. Regular weight updates and redeployment to the detection pipeline enable ongoing performance enhancement, creating a self-improving closed-loop framework.

3. Improved YOLOv8s Defect Detection Algorithm

During algorithm selection for the steel cord ply defect detection system, YOLOv5, YOLOv8, and YOLOv11 models were evaluated using 5-fold cross-validation. YOLOv8s was chosen for further improvement due to its superior training efficiency and accuracy.

Although YOLOv8s achieved strong overall metrics on the steel cord ply defect dataset, two critical limitations were observed: limited accuracy for color-sensitive defects and a high miss rate for small-target defects. To understand these shortcomings, an analysis of the network architecture was conducted, and three underlying design constraints were identified.

First, for color-sensitive defects, the RGB channels in the original YOLOv8s are fused at the earliest convolutional layers. While this accelerates high-level feature extraction, it causes early loss of raw color information. Color-sensitive defects such as wax marker traces and yellowish specks exhibit subtle chromatic differences against the dark rubber background, which are easily diluted when mixed with spatial features in shallow layers.

Second, for small-target defects, the original SPPF module offers limited receptive field expansion and lacks the ability to model long-range spatial dependencies. Defects as small as 16×16 pixels occupy an extremely small portion of the input feature map, making them highly susceptible to background interference and spatial information loss during deep network propagation.

Third, for bounding box localization, the CIoU loss function used in YOLOv8s is sensitive to small center point shifts. For small targets, where the Intersection-over-Union (IoU) fluctuates significantly with minor positional deviations, this sensitivity destabilizes convergence and degrades localization accuracy.

Based on these intuitions, three targeted modifications are proposed to address each limitation, as detailed in the following subsections.

1. The standard convolution at the network input was replaced with RGBV grouped convolution. This module separates the input into R, G, B, and V components, processes each independently with dedicated kernels, and then concatenates the feature maps. This approach preserves and emphasizes chromatic information early in feature encoding.
2. The original SPPF module was replaced with the SPPFCSPC-LSKA module. This module uses SPPCSPC for cross-stage feature fusion through a dual-path architecture. LSKA attention mechanisms are added to each of the four multi-scale fusion branches, adaptively adjusting kernel weights across scales for efficient and discriminative feature aggregation. These changes enlarge the model's receptive field and improve robustness in complex industrial environments.
3. The original CIoU loss function was replaced with WIoU, which dynamically adjusts gradient gains to balance samples of varying quality. This change optimizes the model's localization accuracy and robustness.

Together, these modifications form the improved YOLOv8s steel cord ply defect detection algorithm, which integrates RGBV grouped convolution and multi-scale feature fusion in a unified framework. Figure 4 illustrates the improved network architecture.

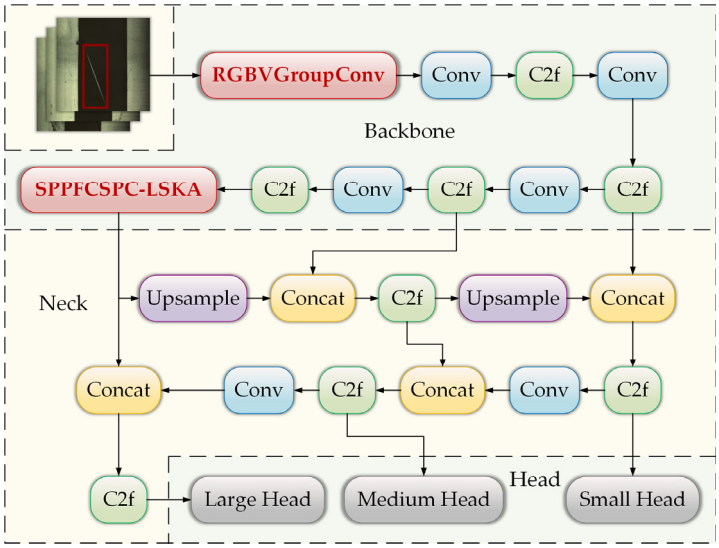


Figure 4. Improved YOLOv8s network architecture.

3.1. Color-Sensitive Defect Detection Algorithm

In steel cord ply defect detection, shape, texture, and size are typically the main distinguishing features. However, for color-sensitive defects with irregular shapes and no clear textural patterns, color attributes are essential. As shown in Figure 5, two common color-sensitive defects are residual brown wax marker traces from manual marking and yellowish specks caused by fine particulate contaminants on the rubber surface.

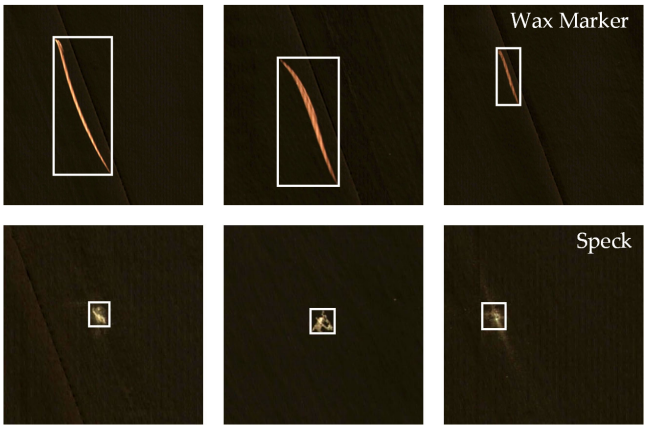


Figure 5. Defect examples of color-sensitive defects. (White box: detected defect region.)

In contrast, defects like localized cord exposure and splice splits have regular shapes and clear textures, making them easier for standard detection algorithms to identify. Wax marker traces result from subjective human actions, while yellowish specks come from random contamination. Both defects show strong color contrast against the dark rubber background, so prioritizing color information is necessary to improve detection.

In the original YOLOv8s architecture, shallow-layer convolutions quickly combine spatial and color information during early feature encoding. While this approach supports efficient extraction of high-level features, it also causes early loss of raw color signals. As the network deepens, abstract representations replace low-level cues, and the initial color information becomes diluted and eventually disappears in deeper layers.

To address the limitations in chromatic information encoding present in the original YOLOv8s architecture, an RGBV grouped convolution module is introduced, as depicted in Figure 6. The core design explicitly separates the R, G, and B chromatic components

to maintain color fidelity and enhances the V luminance channel to increase contrast sensitivity. This dual-mechanism approach improves the discriminative representation of color-sensitive defects.

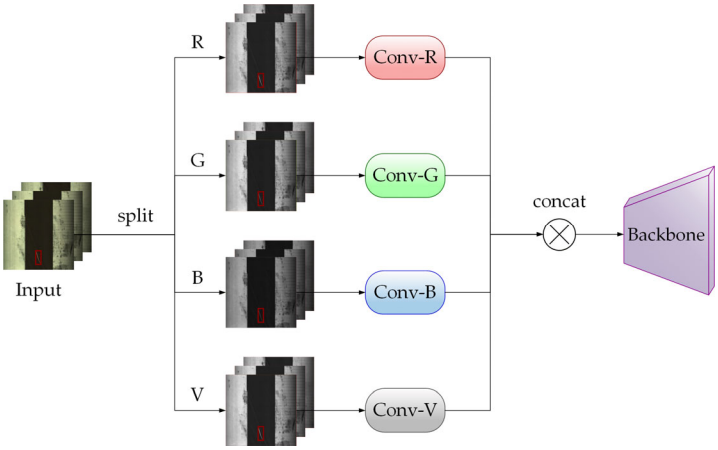


Figure 6. RGBV grouped convolution structure.

Let the input tensor be denoted as $\mathrm{X} \in \mathrm{R}^{B \times C \times H \times W}$. The overall process is illustrated in Equations (1)–(3).

$$\begin{cases} R = \rho_r(\mathrm{X}) \\ G = \rho_g(\mathrm{X}) \\ B = \rho_b(\mathrm{X}) \\ V = \max(R, G, B) \end{cases} \tag{1}$$

In Equation (1), ρ is the channel separation operator. The RGB input is split into three independent chromatic components to enhance color feature extraction. Since steel cord ply defect detection occurs under changing illumination, environmental lighting varies over time. The luminance component V is computed as the element-wise maximum across the R, G, and B channels.

This formulation, while simpler than the standard luminance formula $L = 0.299R + 0.587G + 0.114B$, serves a different purpose in our context. The standard weighted sum approximates human visual perception, which tends to suppress chromatic differences. However, in industrial inspection scenarios, human operators may miss subtle defects due to low color contrast between the defect and the background. Therefore, the algorithm should not simply mimic human perception but rather enhance discriminative features that are visually challenging.

The steel cord ply background consists of dark rubber, for which the RGB components are consistently low across all three channels. In contrast, different defect types exhibit distinct chromatic signatures: wax marker traces appear brownish with higher values in the red and green channels; yellowish specks show elevated responses in the green and red channels; while localized cord exposure or linear cord exposure appear as brighter metallic areas with relatively uniform intensity across channels. The max operation preserves the strongest chromatic response across channels, effectively amplifying color contrast between defects and the dark rubber background. Furthermore, the max operation is computationally lightweight and introduces no additional floating-point operations, which is beneficial for real-time industrial deployment.

$$\begin{cases} F_r = C_r(R) \\ F_g = C_g(G) \\ F_b = C_b(B) \\ F_v = C_v(V) \end{cases} \tag{2}$$

In Equation (2), C denotes the convolution operation. Each group of convolutional kernels processes only its assigned RGBV component, which prevents early mixing of chromatic information in the network's shallow layers. This design ensures that the chromatic characteristics of different defect types, such as the yellowish hue of wax marker traces or the metallic brightness of cord exposure, are preserved in separate feature channels before being combined in deeper layers. By maintaining this separation, the network learns to associate specific chromatic patterns with corresponding defect categories without interference from other color components.

$$Y = \left[F_r \otimes F_g \otimes F_b \otimes F_v\right] \in R^{B\times 4M\times H'\times W'} \quad (3)$$

In Equation (3), $\otimes$ denotes channel concatenation, and M = out_channels/4. The input feature maps undergo grouped convolution, are concatenated by channel to form a refined output tensor, and are then passed to the backbone network for hierarchical feature extraction. This concatenation step reassembles the independently processed chromatic and luminance features, providing the backbone network with a rich representation that retains both color fidelity and spatial details. The unified tensor preserves the enhanced color contrast achieved by the max operation while maintaining the distinctiveness of each chromatic channel, enabling subsequent layers to leverage this information for accurate detection of color-sensitive defects.

3.2. Small-Target Defect Detection Algorithm

Small-target defects are challenging to detect due to their small size, precise localization needs, and vulnerability to complex backgrounds. As shown in Figure 5, yellowish specks in steel cord ply defect detection are dual-modality defects that are both color-sensitive and small, typically measuring 16 × 16 to 32 × 32 pixels. This size is much smaller than the 4096 × 4096 pixels input resolution, leading to a higher false negative rate for these critical defects.

3.2.1. SPPFCSPC-LSKA Module

To address input dimension constraints and support multi-scale feature fusion, He et al. [28] introduced the SPP module in 2015. Its main innovation is a parallel multi-kernel pooling design using different kernel sizes, such as 5 × 5, 9 × 9, and 13 × 13. This approach processes various input dimensions and prevents information loss from image cropping or scaling. However, the parallel architecture increases computational redundancy and reduces inference efficiency.

The SPPF module, introduced by the YOLOv5 authors as an adaptation of SPP, uses a single-kernel serial pooling design shown in Figure 7a. It applies a 5 × 5 pooling layer three times in sequence, with each output feeding into the next stage. The original input features are concatenated with the outputs from all three pooling stages to form the final feature representation. This design reduces computational complexity and improves inference speed. Serial stacking of the 5 × 5 kernel achieves performance similar to parallel multi-kernel pooling with 5 × 5, 9 × 9, and 13 × 13 kernels, maintaining comparable detection accuracy. However, the single-kernel serial approach limits receptive field expansion, reducing its effectiveness for multi-scale feature extraction in complex defect detection.

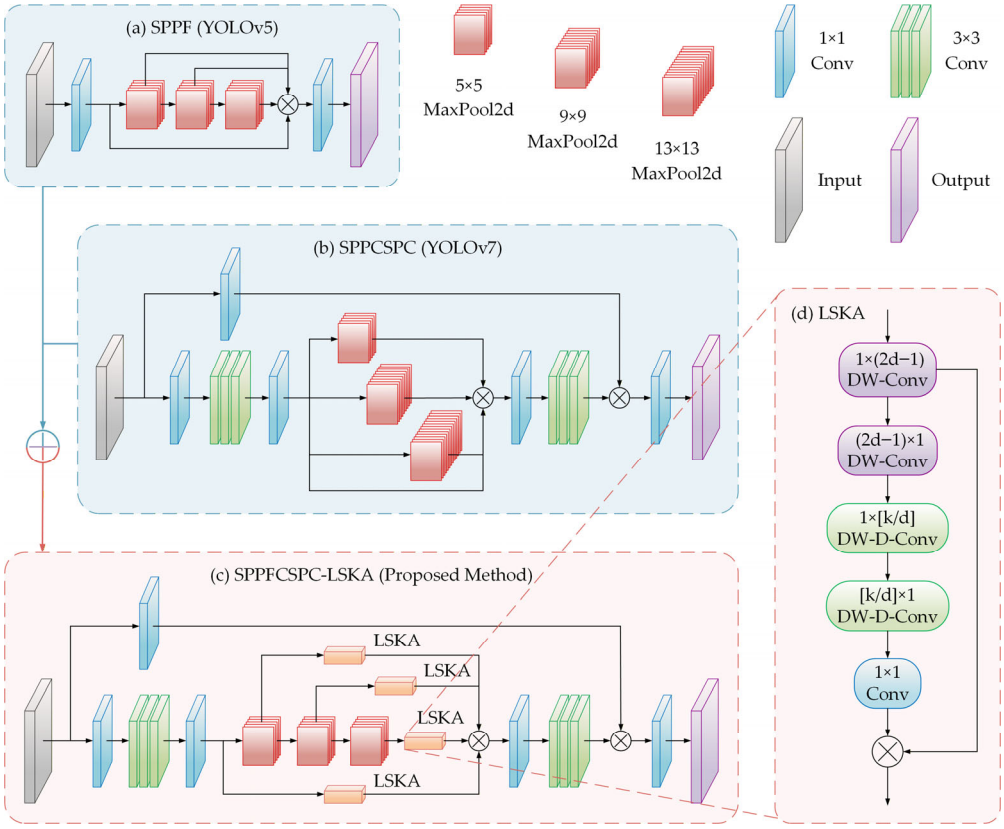


Figure 7. SPPFCSPC-LSKA module design. (⊗: channel concatenation; ⊕: module combination.)

The SPPCSPC module, introduced by the YOLOv7 authors by combining SPP and CSP architectures, uses a dual-branch parallel design shown in Figure 7b. One branch processes input features with standard convolutional layers, while the other extracts multi-scale features using SPP's parallel multi-kernel pooling. Outputs from both branches are concatenated and refined through additional convolutional layers for cross-stage feature fusion. This design helps prevent vanishing gradients and improves contextual feature representation. However, the dual-branch structure and parallel pooling increase model parameters and computational load, slightly reducing inference speed compared to more streamlined designs.

The SPPFCSPC module combines the computational efficiency of SPPF with the cross-stage fusion of SPPCSPC using a dual-branch architecture. One branch processes input features with standard convolutional layers, while the other uses SPPF-inspired triple iterative single-kernel pooling to speed up inference. The concatenated outputs from both branches are fused, reducing computational redundancy while preserving contextual representation. This design maintains feature integrity during fusion and improves sensitivity to small-target defects in steel cord ply defect detection.

Conventional pooling and feature concatenation are not sufficient to enhance spatial information or suppress noise in steel cord ply defect detection with complex backgrounds. To address this, the LSKA attention mechanism is integrated into the SPPFCSPC architecture. As shown in Figure 7d, LSKA uses depthwise large-kernel convolution to mimic self-attention, capturing broad spatial context at lower computational cost than standard large-kernel convolution. Pointwise convolution then performs channel fusion and nonlinear transformation to create a spatial attention map that highlights key image regions. By multiplying this attention map with input features, LSKA achieves spatially adaptive feature modulation. This design establishes long-range spatial dependencies while reducing parameter count and computational complexity.

Previous research has integrated LSKA into the standard SPPF module in two ways: after channel concatenation or before pooling branch fusion. Experiments confirm that SPPF-LSKA suppresses complex backgrounds and improves multi-scale object detection. Building on this, the SPPFCSPC-LSKA module is proposed as a multi-scale fusion mechanism, with its architecture shown in Figure 7c. Scale-specific LSKA attention modules are added to each branch of the serial pooling layers, enabling spatially adaptive refinement of multi-scale features before final concatenation. This design maintains SPPFCSPC's computational efficiency and contextual representation while adding LSKA's long-range contextual modeling. As a result, detection performance for small-target defects in complex industrial environments is improved for steel cord ply defect detection.

3.2.2. WIoU Loss Function

In steel cord ply defect detection, small-target defects occupy little space in feature maps, making their key features vulnerable to background interference and loss of spatial resolution during deep convolutional processing. This increases false negative rates. Minor shifts in the bounding box center cause significant drops in IoU for small targets, while the same shifts have minimal effect on large targets due to scale differences.

The original YOLOv8s model employs Complete IoU (CIoU) as the bounding box regression loss function, defined in Equations (4)–(6):

$$L_{\mathrm{CIoU}} = 1 - R_{\mathrm{IoU}} + \frac{(x - x_{gt})^2 + (y - y_{gt})^2}{W_g^2 + H_g^2} + \alpha v \tag{4}$$

$$v = \frac{4}{\pi^2}\left(\arctan\frac{w}{h} - \arctan\frac{w_{gt}}{h_{gt}}\right)^2 \tag{5}$$

$$\alpha = \frac{v}{1 - R_{\mathrm{IoU}} + v} \tag{6}$$

In these equations, IoU refers to the intersection-over-union between predicted and ground-truth bounding boxes. The variables x and y denote the coordinates of the predicted box center, and x_{gt} and y_{gt} denote the ground-truth box center. W_g and H_g are the width and height of the minimum enclosing rectangle for both boxes. α is the weighting coefficient, v is the aspect ratio consistency metric, w and h are the width and height of the predicted box, and w_{gt} and h_{gt} are the width and height of the ground-truth box.

As shown in Figure 8, the CIoU loss function has key limitations for steel cord ply defect detection. For small targets, minor center point shifts cause large fluctuations in L_{CIoU}, which significantly hinders model convergence stability.

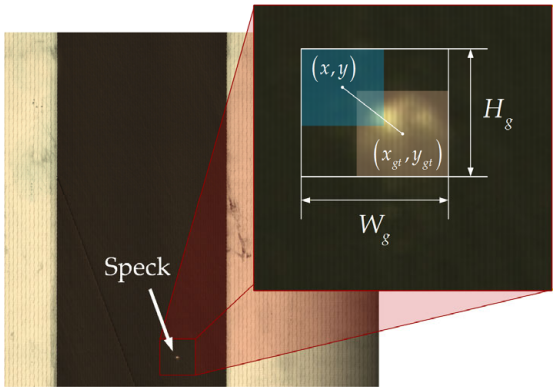


Figure 8. CIoU loss function principle.

The WIoU loss function uses a dynamic focusing mechanism that adjusts gradient gain based on the degree of outlier. High-IoU samples receive increased gradient weights to speed up convergence, while low-IoU samples have reduced weights to limit interference. Its mathematical formulation is given in Equations (7)–(10):

$$L_{\mathrm{WIoU}} = r \cdot R_{\mathrm{WIoU}} \cdot L_{\mathrm{IoU}} \tag{7}$$

$$r = \frac{\beta}{\delta \alpha^{\beta-\delta}} \tag{8}$$

$$R_{\mathrm{WIoU}} = \exp\left(\frac{(x - x_{gt})^2 + (y - y_{gt})^2}{\left(W_g^2 + H_g^2\right)^*}\right) \tag{9}$$

$$\beta = \frac{L_{\mathrm{IoU}}^*}{\overline{L}_{\mathrm{IoU}}} \tag{10}$$

In these equations, r is the gradient gain, α and δ are hyperparameters, β is the outlier degree, and L_{IoU} is the adaptive mean value. The adaptive mean L_{IoU} changes with sample quality: as sample quality declines, β increases and r decreases to suppress harmful gradients. When sample quality improves, β decreases, and r increases to speed up convergence. Small-target samples consistently have lower L_{IoU} values than large-target samples. By dynamically adjusting the gradient gain to reduce interference, WIoU significantly improves overall model performance.

3.3. Knowledge Distillation

3.3.1. MGD Knowledge Distillation

MGD knowledge distillation is a feature-based transfer learning method that improves student model representation using dynamic spatial masking. Instead of replicating teacher outputs, the student reconstructs teacher features with spatially adaptive weighting. The process includes three steps: aligning channel dimensions between student and teacher feature maps, generating spatial attention masks from teacher features with a lightweight generator, and computing weighted distillation loss guided by these masks. This loss is combined with the primary task loss to optimize knowledge transfer.

3.3.2. Multi-Teacher Knowledge Distillation

Enhancing conventional models with defect-category-specific optimization modules often improves individual validation performance. However, combining these modules in a unified architecture can degrade overall performance due to conflicting objectives. Single-teacher knowledge distillation is ineffective in these cases, as it transfers conflicting representations to the student model and limits its potential. To address this, multi-teacher stepwise hierarchical distillation is used, as shown in Figure 9.

Multiple enhanced teacher models, each with specialized modules for specific defect categories, are used to transfer complementary strengths to a single student model. Distillation occurs in sequential, category-specific stages, with each stage focused on features from one teacher. Feature representations are hierarchically aligned at multiple abstraction levels. This phased approach allows the student to progressively acquire specialized capabilities from different teachers while avoiding conflicts. As a result, detection performance improves across various defect types without inheriting inter-module conflicts, creating a coherent representation space.

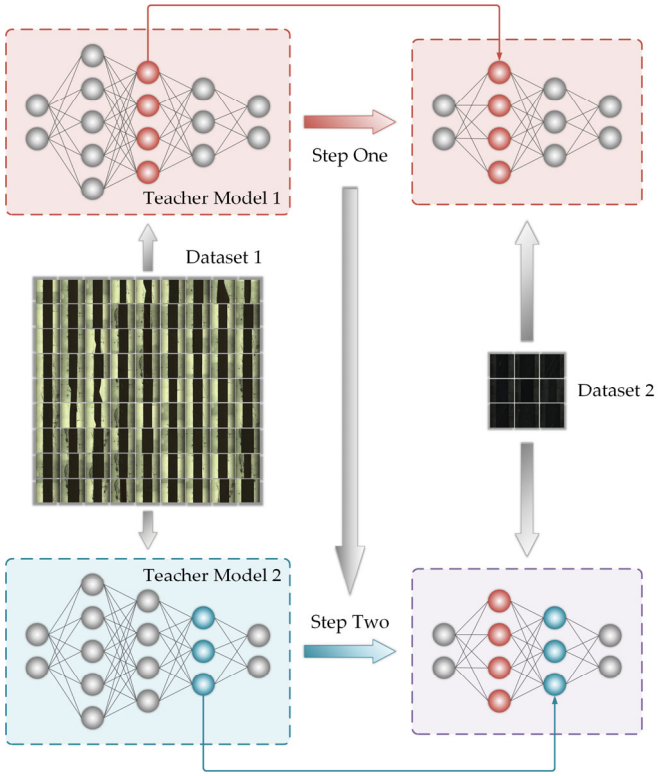


Figure 9. Multi-teacher stepwise hierarchical distillation schematic.

4. Experimental Results and Analysis

All experiments were conducted on the steel cord ply defect datasets described in Section 4.3. During both training and inference, no image resizing or cropping was applied; the original high-resolution images were directly fed into the network. This design preserves the full field of view and retains critical spatial details for small-target defects. The YOLOv8s architecture inherently supports variable input dimensions through its spatial pyramid pooling structures, allowing the network to process images at their native resolution without the need for fixed-size inputs. The training protocol and evaluation metrics are detailed in the following subsections.

4.1. Experimental Setup

All experiments were performed on a Windows 10 system with an Intel Xeon Gold 5218 CPU, 128 GB RAM, and an NVIDIA GeForce RTX 3090 GPU with 24 GB VRAM. Model training used 300 epochs, a batch size of 64, 8 data loading workers, and an image size of 640 pixels.

4.2. Evaluation Metrics

To comprehensively evaluate the performance of the proposed steel cord ply defect detection model, several metrics commonly used in object detection tasks are adopted.

Precision is defined as the proportion of correctly detected defects among all detected targets, reflecting the model's ability to avoid false positives. Recall is defined as the proportion of correctly detected defects among all ground-truth defects, reflecting the model's ability to avoid false negatives. Their definitions are given in Equations (11) and (12):

$$Precision = \frac{TP}{TP + FP} \tag{11}$$

$$Recall = \frac{TP}{TP + FN} \tag{12}$$

In these equations, True Positives (*TP*) denotes the number of defect samples correctly identified, False Positives (*FP*) denotes the number of non-defect samples incorrectly identified as defects, and False Negatives (*FN*) denotes the number of defect samples missed by the model.

*F*1-*score* is computed as the harmonic mean of precision and recall, providing a balanced measure of the model's overall detection performance. It is defined in Equation (13):

$$F1 = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{13}$$

Mean Average Precision (mAP) is employed as the primary metric for evaluating overall detection accuracy. It is computed as the average of Average Precision (AP) across all defect categories, where AP is defined as the area under the precision-recall curve. In this study, mAP is reported at an IoU threshold of 0.5, denoted as mAP@0.5, following common practice in industrial defect detection. This choice facilitates fair comparison with prior works and ensures that localization quality is explicitly considered alongside classification accuracy.

In addition to detection accuracy, model efficiency is also evaluated using four metrics: the number of parameters, Floating-point Operations (FLOPs), inference time, denoted as t_{inf}, and training time, denoted as t_{train}. These metrics are critical for assessing the feasibility of real-time deployment on production lines operating at 50 m/min.

4.3. Data Source

The steel cord ply defect detection system was implemented in Rubber Company A's production environment, with data collected over two months. On average, 32,708 images were captured daily, totaling about 118 GB per day. After filtering, the final dataset included 660 high-resolution images (4096 × 4096 pixels) across eight defect categories: edge notch (30), splice misalignment (230), splice distortion (80), splice split (60), localized cord exposure (150), linear cord exposure (50), speck (40), and wax marker trace (20), as shown in Figure 10. The dataset was split into training, validation, and test sets in an 8:1:1 ratio. All comparative and ablation experiments for the proposed algorithm used this benchmark dataset.

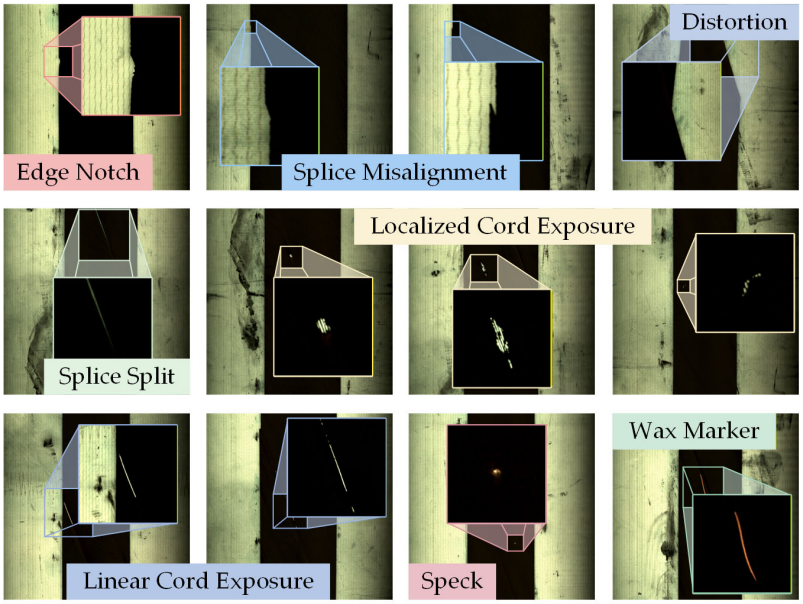


Figure 10. Defect type overview: Production line A.

The system was then deployed on the production line at Tire Manufactory B for one month of data collection. An average of 8393 images were captured daily, totaling 25 GB per day. After curation, the final dataset included 60 high-resolution images (4090 × 4090 pixels) with three defect categories: localized cord exposure (20), linear cord exposure (10), and wax marker trace (30), as shown in Figure 11. The dataset was divided into training, validation, and test sets in an 8:1:1 ratio.

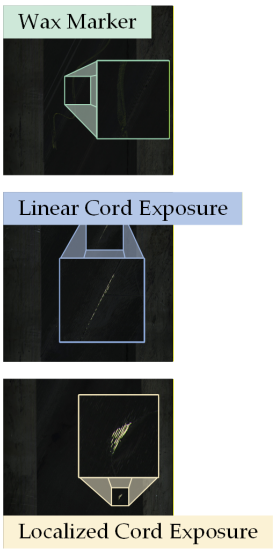


Figure 11. Defect type overview: Production line B.

The production line A dataset contains an uneven class distribution, with certain defect categories, such as wax marker trace, represented by only 20 samples, while splice misalignment has 230 samples. While no explicit balancing strategy was applied, the proposed modules, namely the RGBV grouped convolution and the SPPFCSPC-LSKA module, help improve detection for minority classes. This is reflected in the improved detection performance for minority classes, as will be demonstrated in the following sections.

For both datasets, all original images were directly fed into the network without resizing or cropping. The dataset of production line A consists of 4096 × 4096 images, while the dataset of production line B contains 4090 × 4090 images. The slight difference in dimensions arises from variations in the line-scan camera calibration across production lines. No standardization or resizing was applied to preserve the full field of view and avoid discarding spatial details critical for small-target defects. Consequently, ground-truth bounding box coordinates are also kept at their original scale, requiring no coordinate mapping and introducing no additional localization error from scaling or cropping.

Because the original high resolution is retained, small-target defects occupy an extremely small proportion of the input feature map, making them more susceptible to background interference and spatial information loss during deep network propagation. This design choice further underscores the need for enhanced multi-scale feature extraction mechanisms such as the proposed SPPFCSPC-LSKA module, which, as discussed in Section 3.2, supports variable input resolutions through its multi-scale pooling structure, allowing the network to process high-resolution inputs without resizing.

4.4. Comparative Experiments

4.4.1. Baseline Model Comparison

To justify the choice of YOLOv8s as the baseline architecture, several representative single-stage object detectors from the YOLO family were compared on the production line A dataset under identical training settings. Models evaluated include YOLOv5s, YOLOv8s,

and YOLOv11s. To account for variability in data partitioning, five independent experiments were conducted with different random splits. The mean and standard deviation of precision, recall, F1-score, and mAP@0.5 over the five runs are reported in Table 1, along with training time, FLOPs, and parameter counts.

Table 1. Baseline model comparison results.

Model	P (%)	R (%)	F1-Score (%)	mAP@0.5 (%)	t_{train} (h)	FLOPs (G)	Params (M)
YOLOv5s	85.3 ± 4.7	84.0 ± 1.3	84.6 ± 1.9	87.0 ± 2.0	7.2	**15.8**	**7.034**
YOLOv8s	**88.4** ± 1.7	87.5 ± 2.9	**87.9** ± 1.7	91.5 ± 3.1	**0.714**	28.5	11.129
YOLOv11s	86.8 ± 4.8	**89.0** ± 2.9	87.8 ± 2.6	**93.1** ± 2.3	0.775	21.3	9.416

Bold indicates the best value for each metric.

As shown in Table 1, YOLOv8s achieves a well-balanced performance across all metrics. It attains a mean precision of 88.4% with a standard deviation of 1.7 percentage points, a recall of 87.5% ± 2.9, an F1-score of 87.9% ± 1.7, and an mAP@0.5 of 91.5% ± 3.1. While YOLOv11s achieves the highest mean recall of 89.0% and the highest mean mAP@0.5 of 93.1%, it exhibits slightly lower precision at 86.8% and larger variance in mAP@0.5 with standard deviations of 2.6 percentage points, indicating greater sensitivity to data partition variations. YOLOv5s, despite having the smallest parameter count of 7.034 M and the lowest FLOPs at 15.8 G, demonstrates significantly lower detection accuracy and recall, along with a considerably longer training time of 7.0 h, which hinders rapid model iteration in industrial settings.

In terms of stability across different data splits, YOLOv8s shows the lowest standard deviations in precision and F1-score at 1.7 percentage points, suggesting superior robustness. YOLOv5s and YOLOv11s, on the other hand, exhibit considerable instability, with precision standard deviations of 4.7 and 4.8 percentage points, respectively. YOLOv11s also shows larger fluctuations in mAP@0.5 with standard deviations of 2.6 percentage points. This variation can be attributed to the data scarcity and class imbalance inherent in industrial steel cord ply datasets, where certain defect categories are severely underrepresented. Under such conditions, models with less optimized architectures are more susceptible to overfitting on specific training splits and fail to generalize consistently across different data partitions. YOLOv8s, by contrast, demonstrates a more robust feature representation that maintains stable performance despite data scarcity and imbalance.

Considering the trade-off between detection accuracy, training efficiency, and stability, YOLOv8s strikes the best balance for real-time industrial deployment. Therefore, YOLOv8s was selected as the base model for further architectural enhancements.

4.4.2. Grouped Convolution Replacement Comparison

A systematic evaluation was conducted to assess the positional efficacy of RGBV grouped convolution within the network architecture and to validate its effectiveness when replacing the initial convolutional layer. Comparative experiments on the baseline YOLOv8s architecture integrated the RGBV grouped convolution module at specific locations, with all other components unchanged. Table 2 summarizes the quantitative results.

The results show that replacing only the 1st-stage convolution with RGBV grouped convolution in the baseline architecture achieves 92.4% precision, 91.5% recall, and 96.8% mAP@0.5 on the industrial steel cord ply defect dataset. All metrics improved compared to the baseline. This configuration also reduces inference time from 4.9 ms to 4.2 ms while maintaining comparable computational cost, with FLOPs of 28.3 G versus 28.5 G and parameters of 11.128 M versus 11.129 M. The improvements confirm that RGBV component

separation and fusion at shallow network stages help the model focus on important color features in raw input images.

Table 2. Grouped convolution position comparison results.

Replacement Position	P (%)	R (%)	F1-Score (%)	mAP@0.5 (%)	t_{inf} (ms)	FLOPs (G)	Params (M)
Baseline	88.5	89.8	89.1	93.8	4.9	28.5	11.129
2nd-stage	74.9	**92.8**	82.9	91.6	4.3	27.5	11.111
3rd-stage	88.1	88.1	88.1	93	4.9	27.5	11.056
1st-stage + 2nd-stage	92.2	88.0	90.1	95.8	4.5	**27.4**	11.110
1st-stage + 5th-stage	90.9	87.6	89.2	94.2	**4.2**	**27.4**	**10.835**
1st-stage (proposed)	**92.4**	91.5	**91.9**	**96.8**	**4.2**	28.3	11.128

Bold indicates the best one or two values for each metric.

Replacing convolutional stages in mid-network regions does not improve performance and instead causes measurable declines. As shown in Table 2, substituting the 2nd-stage reduces mAP@0.5 by 2.2 percentage points to 91.6% and reduces precision from 88.5% to 74.9%, a drop of 13.6 percentage points. Replacing the 3rd-stage results in performance similar to the baseline. These results indicate that RGBV operations in mid-network stages disrupt feature abstraction and compromise the model's semantic hierarchy.

Multi-stage replacement strategies offer limited benefits. Adding RGBV grouped convolution to both the 1st and 2nd stages achieves 92.2% precision, 88.0% recall, and 95.8% mAP@0.5, which is better than the 2nd-stage-only setup but worse than the 1st-stage-only variant. Similarly, the 1st + 5th-stage configuration outperforms the baseline but underperforms compared to the 1st-stage-only and 1st + 2nd-stage variants. These results show that adding RGBV operations beyond the 1st-stage introduces interference and reduces the gains from 1st-stage optimization.

In summary, replacing the 1st-stage convolution with RGBV grouped convolution maximizes color feature extraction from raw images and improves detection accuracy for color-sensitive defects in steel cord ply manufacturing, while also reducing inference time and maintaining computational efficiency.

4.4.3. Spatial Pyramid Pooling Comparison

A rigorous ablation study was conducted on the baseline YOLOv8s architecture to evaluate the impact of different SPP variants and SPPFCSPC-LSKA configuration strategies. Benchmark metrics for the standard SPPF, SPPCSPC, and SPPFCSPC modules were compared. Six architectural variants incorporating the LSKA attention mechanism into the SPPFCSPC module were then evaluated, and the optimal configuration was identified through metric analysis. All results are presented in Table 3.

The results show that the SPPCSPC module offers slightly higher precision and mAP@0.5 than the SPPF module in the baseline YOLOv8s architecture, but with lower recall. The SPPCSPC module also reduces inference time from 4.9 ms to 3.9 ms, though it increases parameters by 6.425 M and FLOPs by 5.1 G. The SPPFCSPC module combines the strengths of both, achieving 90.1% precision, 91.8% recall, and 95.8% mAP@0.5, outperforming both previous modules in all detection metrics. It maintains the same parameter count of 17.554 M and FLOPs of 33.6 G as SPPCSPC while improving recall by 3.3 percentage points. As a result, the SPPFCSPC module was chosen to replace the SPPF module and was further optimized.

Table 3. Spatial pyramid pooling comparison results.

SPP Variant	P (%)	R (%)	F1-Score (%)	mAP@0.5 (%)	t_{inf} (ms)	FLOPs (G)	Params (M)
SPPF	88.5	89.8	89.1	93.8	4.9	**28.5**	**11.129**
SPPCSPC	89.2	88.5	88.8	94.8	3.9	33.6	17.554
SPPFCSPC	90.1	91.8	90.9	95.8	4.6	33.6	17.554
SPPF-LSKA	88.8	91.1	89.9	94.4	**2.0**	29.3	12.199
SPPFCSPC-LSKA (after cv6)	82.9	90.0	86.3	95.6	5.8	33.8	17.825
SPPFCSPC-LSKA (after cv1)	84.5	89.1	86.7	92.8	4.0	33.8	17.825
SPPFCSPC-LSKA (before cv5)	85.6	92.1	88.7	95.3	4.4	37.0	21.791
SPPFCSPC-LSKA (before cv7)	84.0	90.9	87.3	93.4	4.3	33.8	17.825
SPPFCSPC-LSKA (dual-branch)	84.6	87.3	85.9	93.9	5.3	34.0	18.096
SPPFCSPC-LSKA (proposed)	**91.1**	**92.5**	**91.8**	**96.5**	3.7	35.3	19.687

Bold indicates the best value for each metric.

The performance of SPPFCSPC-LSKA depends strongly on the integration strategy. Integrating LSKA after CV6 or CV1 increases mAP@0.5 but significantly reduces precision and recall, indicating that attention mechanisms at these positions disrupt key feature propagation. These variants also incur higher computational cost, with the after-cv6 variant increasing inference time to 5.8 ms. Integrating LSKA before cv5 or cv7 provides only slight performance gains and lowers precision compared to the baseline SPPFCSPC module, while the before-cv5 variant substantially increases parameters to 21.791 M and FLOPs to 37.0 G. Applying LSKA to both cross-stage fusion branches at once further reduces performance and increases inference time to 5.3 ms.

The proposed multi-scale fusion mechanism applies LSKA across all four branches of the SPPFCSPC structure, achieving 91.1% precision, 92.5% recall, and 96.5% mAP@0.5 on the industrial steel cord ply defect dataset, surpassing the baseline SPPFCSPC module by 1 percentage point in precision and 0.7 percentage points in mAP@0.5. This method improves recall by 8.5 percentage points for 30 × 30 pixel contaminant defects. In terms of efficiency, the proposed module achieves an inference time of 3.7 ms, which is 0.9 ms faster than the baseline SPPFCSPC module, with a moderate increase in parameters to 19.687 M and FLOPs to 35.3 G. These findings show that effective multi-scale information fusion with LSKA in the SPPFCSPC module significantly improves the YOLOv8s model's detection accuracy for small-target defects while maintaining favorable inference speed.

4.4.4. Loss Function Comparison

To further improve model performance, the YOLOv8s architecture with RGBV grouped convolution and SPPFCSPC-LSKA modules was used as the backbone for comparative experiments with four loss functions. All metrics are provided in Table 4.

The results show that replacing the original YOLOv8s loss function with Focal loss reduces precision, recall, and mAP@0.5 on the industrial steel cord ply defect dataset, achieving 88.2% precision, 89.8% recall, and 94.4% mAP@0.5. This suggests that the hard example focusing mechanism offers limited benefit for models with enhanced feature extraction. Using SIoU or EIoU loss functions also leads to significant drops in precision and recall, with EIoU achieving only 87.1% precision, 86.6% recall, and 93.3% mAP@0.5, likely due to their aspect ratio constraints interfering with multi-scale target optimization. Only the WIoU loss function provides a notable performance increase, achieving 90.4%

precision, 92.0% recall, and 97.2% mAP@0.5, with an F1-score of 91.2%. In terms of efficiency, WIoU achieves an inference time of 3.9 ms, which is 1.7 ms faster than the baseline CIoU and 0.1 ms faster than FocalIoU. This indicates that WIoU's dynamic gradient adjustment works well with the model's multi-scale fusion, effectively managing sample variation in complex scenarios while maintaining favorable inference speed.

Table 4. Loss function comparison results.

Loss Function	P (%)	R (%)	F1-Score (%)	mAP@0.5 (%)	t_{inf} (ms)
Backbone + CIoU	90.0	90.7	90.3	95.0	5.6
Backbone + FocalIoU	88.2	89.8	89	94.4	4.0
Backbone + EIoU	87.1	86.6	86.8	93.3	5.3
Backbone + SIoU	87.9	86.6	87.2	94.3	4.9
Backbone + WIoU (proposed)	**90.4**	**92.0**	**91.2**	**97.2**	**3.9**

Bold indicates the best value for each metric.

4.5. Ablation Experiments

The performance contributions of the RGBV grouped convolution module, SPPFCSPC-LSKA module, and WIoU loss function to the steel cord ply defect detection algorithm were evaluated. The experimental metrics are summarized in Table 5.

Table 5. Ablation study results.

	RGBV	SPPFCSPC-LSKA	WIoU	P (%)	R (%)	F1-Score (%)	mAP@0.5 (%)	t_{inf} (ms)	FLOPs (G)	Params (M)
1	—	—	—	88.5	89.8	89.1	93.8	4.9	28.5	11.129
2	√	—	—	**92.4**	91.5	**91.9**	96.8	4.2	**28.3**	**11.128**
3	—	√	—	91.1	**92.5**	91.8	96.5	**3.7**	35.3	19.687
4	√	√	—	90.0	90.7	90.3	95.0	5.6	35.2	19.686
5	√	√	√	90.4	92.0	91.2	**97.2**	3.9	35.2	19.686

"—" indicates the module is not selected; "√" indicates the module is selected; bold indicates the best value for each metric.

The results show that all three improvement strategies enhance detection performance. The WIoU loss function, in particular, significantly improves bounding box localization by dynamically optimizing regression weights for various samples in complex scenes. Incorporating the RGBV grouped convolution module, SPPFCSPC-LSKA module, and WIoU loss function led to increases in both recall and mAP@0.5.

As shown in Table 5, adding only the RGBV grouped convolution module increased precision by 3.9 percentage points to 92.4% and mAP@0.5 by 3 percentage points to 96.8%, while reducing inference time from 4.9 ms to 4.2 ms with negligible changes in FLOPs and parameters. Adding only the SPPFCSPC-LSKA module increased recall by 2.7 percentage points to 92.5% and mAP@0.5 by 2.7 percentage points to 96.5%, with an inference time of 3.7 ms, but at the cost of increased parameters from 11.129 M to 19.687 M and FLOPs from 28.5 G to 35.3 G. Beyond the overall metrics, the SPPFCSPC-LSKA module delivers substantial gains across different target scales. Compared to the baseline model, the recall for small-target speck defects improves by 8.5 percentage points, while the recall for large-scale defect types, such as splice distortion, improves by 16.7 percentage points. These improvements are primarily attributed to the multi-scale attention mechanism, which enhances feature extraction across varying receptive fields, enabling the model to better capture both fine-grained details and broader structural patterns.

While combining the two modules caused minor metric fluctuations, their enhanced feature extraction supports further optimization with the WIoU loss function. The final model, which includes all three improvements, achieved a precision of 90.4%, a recall of 92.0%, and mAP@0.5 of 97.2% on the industrial steel cord ply defect dataset, with an F1-score of 91.2%. In terms of efficiency, the full model achieves an inference time of 3.9 ms, which is 1.0 ms faster than the baseline, while maintaining FLOPs at 35.2 G and parameters at 19.686 M, representing a moderate increase that remains acceptable for real-time industrial deployment. These results represent increases of 1.9 percentage points in precision, 2.2 percentage points in recall, and 3.4 percentage points in mAP@0.5 over the original YOLOv8s baseline.

4.6. Model Stability Experiments

To evaluate the stability and generalization of the proposed model under different data splits, five-fold cross-validation was conducted on the production line A dataset. The dataset was randomly partitioned into five equal folds, with each fold used as the validation set in turn while the remaining four were used for training. All hyperparameters were kept consistent with the main experiments described in Section 4.1. Five-fold cross-validation was selected as it provides a balanced trade-off between computational cost and estimation stability, given the moderate size of the dataset (660 images). The mean and standard deviation of precision, recall, F1-score, and mAP@0.5 over the five folds are reported in Table 6.

Table 6. Five-fold cross-validation results.

Model	P (%)	R (%)	F1-Score (%)	mAP@0.5 (%)
YOLOv8s (baseline)	88.4 ± 1.7	87.5 ± 2.9	87.9 ± 1.7	91.5 ± 3.1
Proposed	**91.4** ± 1.0	**90.5** ± 1.8	**90.5** ± 0.9	**96.1** ± 0.8

Bold indicates the best value for each metric.

The cross-validation results demonstrate that the proposed model consistently outperforms the baseline across all metrics. It achieves a mean precision of 91.4%, a recall of 90.5%, an F1-score of 90.5%, and mAP@0.5 of 96.1%, representing improvements of 3 percentage points in precision, 3 percentage points in recall, 2.6 percentage points in F1-score, and 4.6 percentage points in mAP@0.5 over the baseline.

In terms of stability, the proposed model exhibits standard deviations of 1 percentage point for precision, 1.8 percentage points for recall, 0.9 percentage points for F1-score, and 0.8 percentage points for mAP@0.5. Compared to the baseline, which shows standard deviations of 1.7, 2.9, 1.7, and 3.1 percentage points, respectively, the proposed model demonstrates consistently lower variance across all metrics, particularly in recall and mAP@0.5. This indicates that the proposed architectural enhancements not only improve detection accuracy but also enhance model robustness to variations in data partitioning.

The reduced variance is especially notable for mAP@0.5, where the baseline exhibits a standard deviation of 3.1 percentage points while the proposed model achieves 0.8 percentage points. This suggests that the integrated modules, namely the RGBV grouped convolution, the SPPFCSPC-LSKA module, and the WIoU loss function, collectively contribute to a more stable and generalizable feature representation, which is critical for reliable deployment across different production lines where data distributions may vary.

4.7. Comparison with Different Base Networks

To further validate the effectiveness and generalizability of the proposed architectural enhancements, a comparative study was conducted against a diverse set of base networks

within the YOLO family. The selected models cover a wide range of design choices, including different model capacities, multi-scale feature extraction strategies, and lightweight backbone architectures. Specifically, the following models were included.

1. YOLOv5s, YOLOv8s, and YOLOv11s to represent the evolution of the YOLO series, providing a baseline across adopted versions with balanced accuracy and speed.
2. YOLOv8n (nano) and YOLOv8m (medium) to evaluate scalability across different model capacities, from lightweight to more parameter-heavy variants.
3. YOLOv8s-P2 and YOLOv8s-P6 to assess the impact of multi-scale feature pyramids, which are particularly relevant for small-target detection.
4. YOLOv8s-Ghost to test the generalizability of the proposed improvements in resource-constrained scenarios where a lightweight GhostNet backbone is employed.

All models were trained on the production line A dataset under the same training configuration described in Section 4.1. The results are summarized in Table 7.

Table 7. Performance comparison with different base networks.

Model	P (%)	R (%)	F1-Score (%)	mAP@0.5 (%)	t_{inf} (ms)	FLOPs (G)	Params (M)
YOLOv5s	85.1	85.3	87.6	85.2	6.1	15.8	7.034
YOLOv8s	88.5	89.8	89.1	93.8	4.9	28.5	11.129
YOLOv11s	87.3	86.8	87.0	93.9	5.0	21.3	9.416
YOLOv8n	73.4	85.1	78.8	85.5	**0.8**	**8.1**	**3.007**
YOLOv8m	89.4	88.0	88.7	95.3	3.8	78.7	25.844
YOLOv8s-P2	75.8	88.8	81.8	93.5	2.6	36.7	10.629
YOLOv8s-P6	77.5	86.1	81.6	92.0	2.1	28.4	17.860
YOLOv8s-Ghost	72.6	90.2	80.4	92.7	1.6	16.1	5.923
Proposed	**90.4**	**92.0**	**91.2**	**97.2**	3.9	35.2	19.686

Bold indicates the best value for each metric.

As shown in Table 7, the proposed model consistently achieves the highest precision, recall, F1-score, and mAP@0.5 among all compared methods. It attains a precision of 90.4%, a recall of 92.0%, an F1-score of 91.2%, and mAP@0.5 of 97.2%, representing improvements of 1.9 percentage points in precision, 2.2 percentage points in recall, 2.1 percentage points in F1-score, and 3.4 percentage points in mAP@0.5 over the baseline YOLOv8s.

In terms of efficiency, the proposed model achieves an inference time of 3.9 ms, which is 1.0 ms faster than the baseline, while maintaining a moderate parameter count of 19.686 M and FLOPs of 35.2 G. Although the lightweight variants such as YOLOv8n and YOLOv8s-Ghost achieve faster inference speeds of 0.8 ms and 1.6 ms respectively and lower parameter counts, they suffer from significantly lower detection accuracy, with mAP@0.5 values of 85.5% and 92.0% respectively. Conversely, the larger YOLOv8m achieves comparable precision of 89.4% and mAP@0.5 of 95.3% but requires substantially more computational resources, with FLOPs of 78.7 G.

The multi-scale variants YOLOv8s-P2 and YOLOv8s-P6, which incorporate finer or coarser feature pyramids, achieve mAP@0.5 values of 93.5% and 92.0%, respectively, with inference times of 2.6 ms and 2.1 ms. While these variants offer competitive inference speeds, their detection accuracy is lower than the proposed model, which integrates adaptive attention across all four scales within the SPPFCSPC-LSKA module. This confirms that the proposed multi-scale fusion mechanism is more effective than fixed-scale pyramid designs for detecting small-target defects in complex industrial backgrounds.

Overall, the proposed model strikes the most favorable balance between detection accuracy and computational efficiency, outperforming all compared base networks while maintaining inference speeds suitable for real-time industrial deployment. These results

demonstrate that the proposed architectural improvements generalize effectively across different base networks and deliver superior detection performance.

4.8. Public Dataset Experiments

Because color-sensitive defects in steel cord ply occur infrequently in actual production, wax marker trace and speck defects represent only 3.0% and 6.0% of the real measured dataset collected over two months. To further assess the generalization and effectiveness of the improved YOLOv8s model for surface defect detection, additional experiments were performed using the NEU-DET. [29] from Northeastern University. This dataset comprises 1800 images and includes six common steel surface defect types: crack, inclusion, patches, pitted surface, rolled-in scale, and scratches. The dataset was partitioned into training, validation, and test sets in an 8:1:1 ratio prior to training.

The original and improved YOLOv8s models were trained and evaluated separately. The improved model achieved precision, recall, and mAP@0.5 of 72.8%, 74.4%, and 76.6%, respectively, compared to 71.8%, 70.4%, and 75.6% for the original model. The most notable improvement was in recall, confirming the effectiveness of the modifications.

4.9. Knowledge Distillation Experiments

Traditional methods for building comprehensive defect datasets often take over six months, which delays improvements in system accuracy. However, enterprises require rapid deployment and application, rather than waiting for long-term data accumulation to achieve accuracy.

For the knowledge distillation experiment, the teacher model is an improved YOLOv8 defect detection model trained on the production line A dataset, which includes 8 defect types and demonstrates high accuracy and strong feature representation. The student model is trained on a smaller dataset from production line B, containing only three defect types, to simulate data scarcity during production line transfer.

Although defect texture and image clarity are similar between the two production lines, significant differences exist in brightness distribution and overall color features. Furthermore, variations in worker practices result in different trajectories and color saturation of wax marker trace defects in production line B compared to the teacher model's dataset.

4.9.1. Experimental Results (Single-Teacher)

The MGD knowledge distillation method was applied across multiple production lines to assess its transferability. This experiment evaluated whether knowledge distillation can improve model performance when data are limited. Experimental results are shown in Table 8.

Table 8. Knowledge distillation performance.

Model	Dataset	P (%)	R (%)	F1-Score (%)	mAP@0.5 (%)	t_{inf} (ms)	FLOPs (G)	Params (M)
Teacher Model	660	**90.4**	**92.0**	**91.2**	**97.2**	**3.9**	35.2	19.686
Initial Student Model	60	51.2	66.7	57.9	67.4	5.0	**28.4**	**11.127**
Distilled Student Model	60	84.6	82.0	83.3	88.8	11.3	**28.4**	**11.127**

Bold indicates the best one or two values for each metric.

The teacher model, trained on 8 defect categories, achieved high precision, recall, and mAP@0.5. In contrast, the student model trained on a small dataset with 3 categories showed limited performance, with precision, recall, and mAP@0.5 at 51.2%, 66.7%, and 67.4%, respectively. This highlights the challenge of feature extraction under data scarcity. After applying MGD knowledge distillation, the student model's precision improved

by 33.4 percentage points to 84.6%, recall improved by 15.3 percentage points to 82.0%, and mAP@0.5 improved by 21.4 percentage points to 88.8%. These results demonstrate that knowledge distillation transfers knowledge from the teacher to the student model, allowing the student model to approach the teacher's accuracy while maintaining the same parameter count of 11.127 M and FLOPs of 28.4 G as the initial student model. Although inference time increases from 5.0 ms to 11.3 ms due to the additional distillation process, the compact student model remains suitable for real-time industrial deployment, meeting the requirements for rapid deployment and high precision in production line transfers.

4.9.2. Deployment Verification Results (Single-Teacher)

After deploying the teacher model weights to the defect detection system, high accuracy achieved on production line A did not transfer to production line B. In the new environment, model performance declined, with frequent false positives, false negatives, and misclassifications, as shown in Figure 12. This decline was mainly due to environmental differences between production lines, including lighting and equipment variations. As a result, the model lacked sufficient generalization capability and could not maintain its original accuracy.

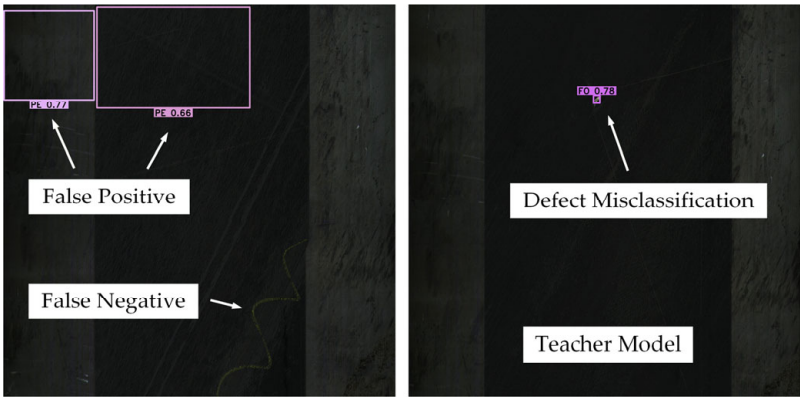


Figure 12. Detection results from teacher model deployment.

Figure 13 shows the deployment results for the student model weights. Using pre-distillation weights, the model performed poorly in the new production environment due to an insufficient number of samples. Frequent false positives occurred, with normal steel cord ply or background misclassified as defects. Statistical analysis also showed severe false negatives, especially low recognition rates for small defects such as localized cord exposure.

After applying MGD knowledge distillation, the student model's performance improved significantly. False positives were nearly eliminated, and false negatives were greatly reduced. This result confirms the effectiveness of knowledge distillation for feature transfer and generalization, and demonstrates its potential for rapid cross-production-line deployment in industrial settings. This provides reliable technical support for rapid production line deployment.

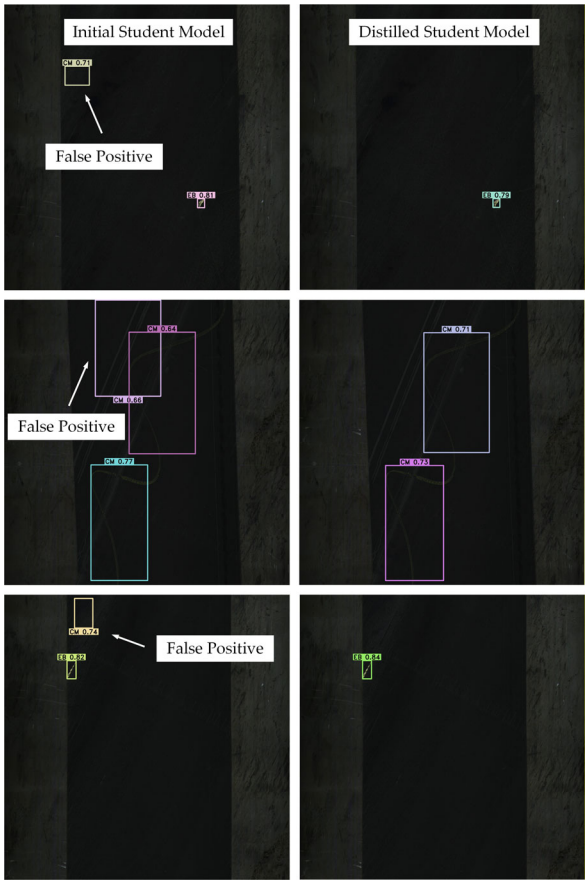


Figure 13. Detection results from student model deployment.

4.10. Multi-Teacher Knowledge Distillation Experiments

Preliminary ablation experiments indicated that replacing either the RGBV grouped convolution module or the SPPFCSPC-LSKA module individually enhanced model performance. In contrast, the simultaneous introduction of both modules resulted in a slight decline in performance. To address this issue, a stepwise hierarchical distillation scheme utilizing multiple teacher models was implemented, and experiments were conducted.

4.10.1. Experimental Results (Multi-Teacher)

In the improved knowledge distillation experiment, both teacher models were trained on the production line A dataset (660 images): one with only the RGBV grouped convolution module replaced, and the other with only the SPPFCSPC-LSKA module replaced. The student model was trained on a small sample dataset from production line B (60 images). The scheme consisted of two steps. First, Teacher Model 1 trained the original student model using the MGD knowledge distillation method. Second, the resulting student model was further trained by Teacher Model 2 in a second distillation step. Experimental data are provided in Table 9.

Each distillation step constitutes a complete training cycle of 300 epochs, rather than a phased subdivision within a single training session. Both steps use the same small-sample dataset from production line B as the training data for the student model. This design ensures that each teacher model has sufficient opportunity to fully transfer its specialized feature representations to the student, without interference from the other teacher. The phased, two-stage approach enables the student to sequentially acquire complementary knowledge from both teachers, effectively resolving the optimization conflicts that arise when both modules are integrated into a single model.

Table 9. Multi-teacher knowledge distillation performance.

Model	Dataset	P (%)	R (%)	F1-Score (%)	mAP@0.5 (%)	t_{inf} (ms)	FLOPs (G)	Params (M)
Teacher Model 1	660	92.4	91.5	91.9	**96.8**	4.2	**28.3**	**11.128**
Teacher Model 2	660	91.1	**92.5**	91.8	96.5	**3.7**	35.3	19.687
Initial Student Model	60	51.2	66.7	57.9	67.4	5.0	28.4	11.127
Student Model 1	60	94.0	86.6	90.1	92.1	4.5	28.4	11.127
Student Model 2	60	**97.5**	88.8	**92.9**	94.3	4.1	28.4	11.127

Bold indicates the best value for each metric.

Experimental results show that after the first-step MGD distillation, the student model's performance improved significantly. The mAP@0.5 increased from 67.4% to 92.1%, a gain of 24.7 percentage points, while precision reached 94.0%, an increase of 42.8 percentage points. After the second-step distillation, overall performance was further optimized, with mAP@0.5 reaching 94.3% and precision reaching 97.5%, surpassing the teacher models' precision of 92.2% and recall of 91.1%. In terms of efficiency, the final student model achieves an inference time of 4.1 ms, which is 0.9 ms faster than the initial student model, while maintaining the same parameter count of 11.127 M and FLOPs of 28.4 G. All detection metrics also exceeded those from previous single-teacher distillation experiments. These results demonstrate that the stepwise hierarchical distillation scheme with multiple teachers is effective. The second-step distillation did not cause interference between modules but integrated the advantages of the second teacher model, resulting in complementary performance and improved generalization beyond single teacher models.

Notably, after multi-teacher distillation, the student model achieved 97.5% precision on the production line B dataset, which contains only 60 training images. This unusually high performance can be attributed to two factors. First, the defect categories in production line B, namely localized cord exposure, linear cord exposure, and wax marker trace, are a subset of those present in production line A, allowing effective knowledge transfer. Second, the stepwise distillation strategy enables the student to inherit well-optimized feature representations from two specialized teachers, thereby achieving strong generalization even with limited samples.

4.10.2. Deployment Verification Results (Multi-Teacher)

To facilitate a clear comparison between single-teacher and multi-teacher distillation modes, the trained model weights were deployed in the defect detection system. The detection results are presented in Figure 14.

Models trained with single-teacher knowledge distillation showed significant improvement in detection accuracy over the original student model. However, they still produced false negatives for some small-target defects and misclassified certain color-sensitive defects. These models also tended to misidentify reflections on the steel cord ply surface or dirt on the conveyor belt as defects. In contrast, models trained with multi-teacher stepwise hierarchical distillation achieved higher defect recognition accuracy. Detection instability was reduced, and overlapping detection boxes were not observed. These results confirm the practical effectiveness of the improved distillation scheme.

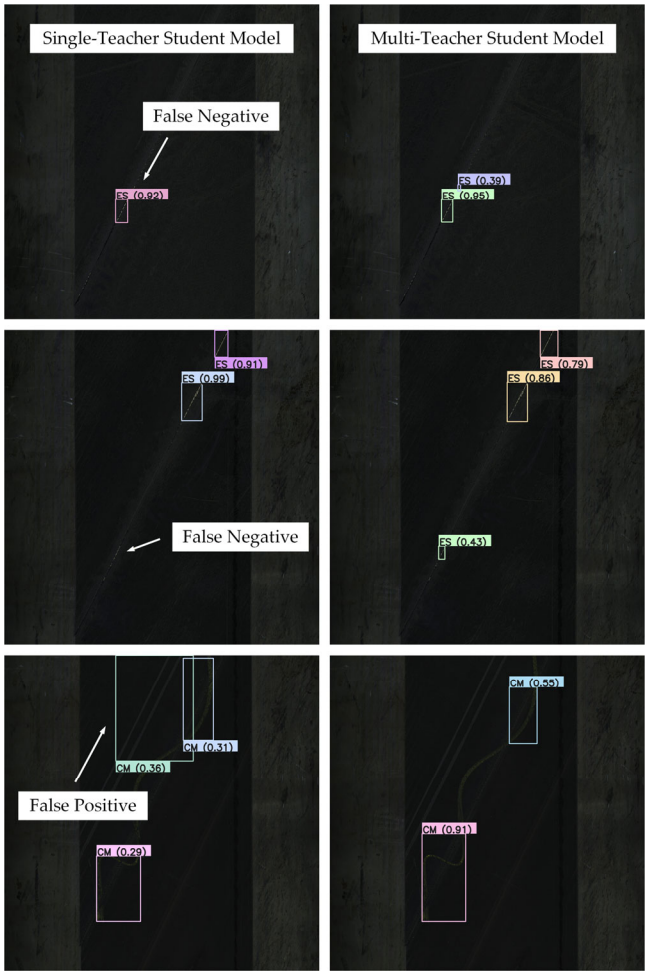


Figure 14. Detection results from multi-teacher knowledge distillation.

5. Conclusions

This study investigates the challenges of low detection accuracy for color-sensitive defects, high miss-detection rates for small-target defects, and limited cross-production-line adaptability in steel cord ply defect detection. An enhanced YOLOv8s detection algorithm is introduced that incorporates knowledge distillation for small-sample cross-production-line scenarios. At the architectural level, RGBV grouped convolution is implemented to improve color feature extraction, the SPPFCSPC-LSKA module is utilized to expand the receptive field for superior small-target defect detection, and the WIoU loss function is integrated to refine localization accuracy. The proposed model achieves a mAP@0.5 of 97.2% on the industrial steel cord ply defect dataset, representing an improvement of 3.4 percentage points over the original YOLOv8s baseline. Notably, wax marker defect detection accuracy increases by 20 percentage points, and the recall rate for 30 × 30 pixels speck defects improves by 8.5 percentage points, demonstrating the model's effectiveness in detecting color-sensitive and small-target defects. Five-fold cross-validation further confirms the model's stability, with low standard deviations across all metrics, and comparisons with different base networks show that the proposed method consistently outperforms alternative architectures.

To improve adaptability across production lines, knowledge distillation is introduced. The high-accuracy improved model acts as a teacher to guide the student model in new production lines with limited samples, addressing issues of insufficient data and weak generalization. A stepwise hierarchical distillation strategy using multiple teacher models is also proposed. Models with only the RGBV grouped convolution or only the SPPFCSPC-

LSKA module serve as teachers, providing phased guidance for the student model to learn enhanced feature representations. This approach avoids module conflicts, integrates their strengths, and enables the student model to outperform single-teacher knowledge distillation in small-sample scenarios, achieving a precision of 97.5%, recall of 88.8%, F1-score of 92.9%, and mAP@0.5 of 94.3% on the target production line dataset. Notably, the distilled student model maintains the same parameter count of 11.127 M and FLOPs of 28.4 G as the initial student model, with an inference time of 4.1 ms, demonstrating that high accuracy can be achieved without sacrificing deployment efficiency.

While the proposed method performs well in steel cord ply defect detection, further improvements remain possible. First, model pruning and quantization could accelerate inference and reduce model size. Second, integrating multimodal data like infrared and X-ray could create a fused detection system, improving recognition of challenging defects. Third, combining defect quantification with root cause inference could bridge detection and analysis, providing deeper decision support for production process optimization. Fourth, the multi-teacher distillation framework can be extended to incorporate teachers with complementary expertise beyond the current RGBV grouped convolution and SPPFCSPC-LSKA modules. For instance, one teacher could specialize in capturing object location and scene context, another in fine-grained texture analysis, and a third in color-sensitive defect classification. Such a diversified teacher ensemble would enable the student model to acquire a more comprehensive understanding of complex industrial scenes, further enhancing its generalization capability and robustness in real-world deployment.

Author Contributions: Conceptualization, P.H. and Z.X.; methodology, P.H.; software, R.L.; validation, P.H., Z.X., and R.L.; formal analysis, F.Z.; investigation, X.Z.; resources, G.H.; data curation, Z.K.; writing—original draft preparation, P.H.; writing—review and editing, Z.X.; visualization, R.L.; supervision, F.Z.; project administration, P.H.; funding acquisition, G.H. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Guangxi Key Research and Development Program, grant number FN2600640602 and the Guangxi Science and Technology Program, grant number AB22035041.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest: Author Guangzhan Huang was employed by the company Guilin Zhonghao Lichuang Mechanical & Electrical Equipment Company. The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

References

1. Zhou, S.; Chen, L.; Li, Y.; Sun, P.; Su, G.; Meng, J. Study on the influence of different tire cord structures on tire grounding characteristics. *J. Eng. Fiber Fabr.* **2022**, *17*, 15589250221138549. [CrossRef]
2. Jiang, Z.; Fu, J.; Zeng, T.; Liu, R.; Cong, P.; Miao, J.; Sun, Y. Defect R-CNN: A Novel High-Precision Method for CT Image Defect Detection. *Appl. Sci.* **2025**, *15*, 4825. [CrossRef]
3. Tan, L.; Wu, H.; Xu, Z.; Xia, J. Multi-object garbage image detection algorithm based on SP-SSD. *Expert Syst. Appl.* **2025**, *263*, 125773. [CrossRef]
4. Ayachi, R.; Said, Y.; Afif, M.; Alshammari, A.; Hleili, M.; Ben Abdelali, A. Assessing YOLO models for real-time object detection in urban environments for advanced driver-assistance systems (ADAS). *Alex. Eng. J.* **2025**, *123*, 530–549. [CrossRef]
5. Xiao, D.; Wang, H.; Liu, Y.; Li, W.; Li, H. DHSW-YOLO: A duck flock daily behavior recognition model adaptable to bright and dark conditions. *Comput. Electron. Agric.* **2024**, *225*, 109281. [CrossRef]

6. Zhao, Q.; Guo, Q. DAQC-YOLO: A comprehensive optimization strategy for inner-surface defect detection in solar quartz crucibles. *Measurement* **2026**, *257*, 118771. [CrossRef]
7. Wu, W.; Cheng, H.; Pan, J.; Zhong, L.; Zhang, Q. Wavelet-Enhanced YOLO for Intelligent Detection of Welding Defects in X-Ray Films. *Appl. Sci.* **2025**, *15*, 4586. [CrossRef]
8. Njima, W.; Bazzi, A.; Chafii, M. DNN-Based Indoor Localization Under Limited Dataset Using GANs and Semi-Supervised Learning. *IEEE Access* **2022**, *10*, 69896–69909. [CrossRef]
9. Saleh, R.; Ertunç, H. Attention-based deep learning for tire defect detection: Fusing local and global features in an industrial case study. *Expert Syst. Appl.* **2025**, *269*, 126473. [CrossRef]
10. Huang, J.; Huang, Z.; Li, C.; Liu, J. Research on defect detection method of motor control board based on image processing. *J. Comput. Appl. Math.* **2026**, *473*, 116916. [CrossRef]
11. Lu, Y.; Cheng, C.; Zhao, D.; Zhu, D.; Gao, Q. Infrared small target detection algorithm based on nested FPN and interference suppression. *Expert Syst. Appl.* **2025**, *274*, 127029. [CrossRef]
12. Qian, K.; Zou, L.; Wang, Z.; Wang, W. Metallic surface defect recognition network based on global feature aggregation and dual context decoupled head. *Appl. Soft. Comput.* **2024**, *158*, 111589. [CrossRef]
13. Dong, S.; Xu, W.; Zhang, H.; Gong, L. Cot-DCN-YOLO: Self-attention-enhancing YOLOv8s for detecting garbage bins in urban street view images. *Egypt. J. Remote Sens. Space Sci.* **2025**, *28*, 89–98. [CrossRef]
14. Wang, S.; Lu, H.; Ge, C.; Wang, J. FIDC-YOLOv5: A model for personnel fall detection in dense crowds. *Measurement* **2026**, *258*, 119095. [CrossRef]
15. Zhao, B.; Chen, Y.; Jia, X.; Ma, T. Steel surface defect detection algorithm in complex background scenarios. *Measurement* **2024**, *237*, 115189. [CrossRef]
16. Zhou, X.; Wei, W.; Huang, Z.; Su, Z. Study on the Detection Mechanism of Multi-Class Foreign Fiber under Semi-Supervised Learning. *Appl. Sci.* **2024**, *14*, 5246. [CrossRef]
17. Heng, G.; Luo, C.; Sun, K.; Huang, S.; Huang, L. SGBM_YOLO: A high-precision obstacle detection algorithm for assistive navigation based on stereo vision and spatial attention mechanism. *Appl. Soft. Comput.* **2026**, *186*, 114144. [CrossRef]
18. Shi, H.; Jin, Z.; Tang, W.; Wang, J.; Jiang, K.; Xu, M.; Xia, W.; Xu, X. High-performance one-stage detector for SiC crystal defects based on convolutional neural network. *Knowl.-Based Syst.* **2023**, *280*, 110994. [CrossRef]
19. Hao, J.; Sun, X.; Liu, X.; Hua, D.; Hu, J. A lightweight and explainable model for driver abnormal behavior recognition. *Eng. Appl. Artif. Intell.* **2025**, *139*, 109559. [CrossRef]
20. Tie, J.; Zhu, C.; Zheng, L.; Wang, H.; Ruan, C.; Wu, M.; Xu, K.; Liu, J. LSKA-YOLOv8: A lightweight steel surface defect detection algorithm based on YOLOv8 improvement. *Alex. Eng. J.* **2024**, *109*, 201–212. [CrossRef]
21. Xu, J.; Yang, S.; Liang, Q.; Zheng, Z.; Ren, L.; Fu, H.; Yang, P.; Xie, W.; Yang, D. Transillumination imaging for detection of stress cracks in maize kernels using modified YOLOv8 after pruning and knowledge distillation. *Comput. Electron. Agric.* **2025**, *231*, 109959. [CrossRef]
22. Liu, W.; Li, Y.; Huang, S.; Yu, Q. A small target detection algorithm for unmanned aerial vehicles incorporating global information modeling and multi-scale feature interaction. *Digit. Signal Prog.* **2026**, *168*, 105465. [CrossRef]
23. Zhang, F.; Leong, L.; Yen, K.; Zhang, Y. An enhanced lightweight model for small-scale pedestrian detection based on YOLOv8s. *Digit. Signal Prog.* **2025**, *156*, 104866. [CrossRef]
24. Cardellicchio, A.; Nitti, M.; Patruno, C.; Mosca, N.; di Summa, M.; Stella, E.; Reno, V. Automatic quality control of aluminium parts welds based on 3D data and artificial intelligence. *J. Intell. Manuf.* **2024**, *35*, 1629–1648. [CrossRef]
25. He, X.; Zhang, Y.; Zhan, Q. AIN-YOLO: A lightweight YOLO network with attention-based InceptionNext and knowledge distillation for underwater object detection. *Adv. Eng. Inform.* **2025**, *66*, 103504. [CrossRef]
26. Xu, J.; Wang, S.; Han, R.; Wu, X.; Zhao, D.; Zeng, X.; Yin, R.; Han, Z.; Liu, Y.; Sheng, S. Crack segmentation and quantification in concrete structures using a lightweight YOLO model based on pruning and knowledge distillation. *Expert Syst. Appl.* **2025**, *283*, 127834. [CrossRef]
27. Yousefimehr, B.; Ghatee, M.; Razavi-Far, R. Multi-teacher knowledge distillation framework for lightweight anomaly detection. *Neural Netw.* **2026**, *195*, 108267. [CrossRef]
28. He, K.; Zhang, X.; Ren, S.; Sun, J. Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2015**, *37*, 1904–1916. [CrossRef]
29. He, Y.; Song, K.; Meng, Q.; Yan, Y. An End-to-End Steel Surface Defect Detection Approach via Fusing Multiple Hierarchical Features. *IEEE Trans. Instrum. Meas.* **2020**, *69*, 1493–1504. [CrossRef]

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