2025 International Conference on Intelligent Computing
July 26-29, Ningbo, China
https://www.ic-icc.cn/2025/index.php

DASC-YOLO: An Attention Scale-aware Framework for Real-time Leather Defect Detection with Limited Samples

Zihao Li[1,2], Zuohao Wu[3], Hongyu Ao[4], Mingsheng Shang[1] and Guang Li[1,*]

[1] Chongqing Institute of Green and Intelligent Technology, Chinese Academy of Sciences, Chongqing 400714, China
[2] Chongqing School, University of Chinese Academy of Sciences, Chongqing 400714, China
[3] School of Computer Science and Technology, Chongqing University of Posts and Telecommunications, Chongqing 400065, China
[4] School of Information and Software Engineering, University of Electronic Science and Technology of China, Chengdu 610054, China
liguang@cigit.ac.cn

Abstract. Surface defect detection in leather manufacturing faces challenges including multi-scale defects, scarce samples, and texture interference. This study proposes an optimized YOLOv11 framework integrating attention mechanisms, cross-scale feature fusion, and few-shot learning. The backbone network employs spatial-channel attention and feature redundancy reduction to enhance defect discrimination. A cross-scale attention mechanism adaptively fuses multi-resolution features for improved small defect detection. A ProtoNet module addresses sample scarcity while ensuring localization precision. Evaluations on industrial leather datasets and public benchmarks demonstrate the model's effectiveness, achieving 81.0% mAP on leather defects and 79.2% on steel surfaces with 3.02M parameters and real-time inference (70.3 FPS). The framework outperforms conventional methods in accuracy and robustness, offering a practical solution for automated quality inspection in texture-rich industrial scenarios.

Keywords: Surface Defect Detection, Tiny Target, Convolutional Neural Network, Deep Learning.

1 Introductionx

Leather quality impacts product performance and commercial value across industries. Processing-induced defects like scratches, spots, cracks and wrinkles compromise aesthetics, durability and functionality [1]. Current leather surface defect detection rely on inefficient manual visual checks, while experienced inspectors can identify complex defects, these approaches suffer from low productivity, subjective standards, and worker fatigue [2], resulting in inconsistent defect detection [1, 3].

Recently, the rapid advancements in computer vision and deep learning technologies have been notable, some traditional machine vision methods, such as Convolutional Neural Network (CNN)-based defect detection algorithms have demonstrated notable success in identifying surface defects in various industries such as steel [4, 5],

textile [6, 7], and electronics [8], providing new ideas and inspiration for leather defect detection [1, 9, 10].

However, current methodologies face several critical challenges in leather defect detection applications. First, the scarcity of defect samples limits training data and undermines deep learning model performance [1, 11, 12]. Second, identifying minute defects in high-resolution leather detection images presents a significant technical challenge, as models often struggle to detect these defects due to their minuscule size relative to the total surface area. [1]. Third, leather defects exhibit significant variations in morphology and size even within the same category, while some defects exhibit similarities to the intricate texture of leather backgrounds [13, 14]. The cross-scale recognition difficulty primarily stems from the combination of significant variations in defect sizes and the high resolution of the images, which together pose a considerable challenge.

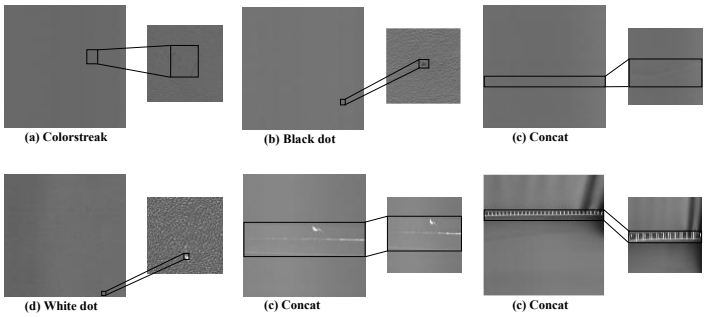

Fig. 1. Typical Defect Examples. Four typical leather defects were selected for this study. Defects (b) and (d) display a significant discrepancy in scale relative to the image dimensions, occupying minimal regions of the visual field. Defect (c) demonstrates pronounced intra-defect size heterogeneity. Defects (a) and (c) exhibit relatively low contrast against the background

To address the identified challenges, we introduce a surface defect detection approach utilizing YOLO11, which incorporates attention mechanisms alongside feature optimization techniques. We adopt YOLO11 as our defect detection model due to its anchor-free architecture and dual optimization through enhanced C3K2 modules with global context awareness and C2PSA components with position-sensitive attention, which collectively enable robust multi-scale defect localization. To further enhance this framework, we integrate the Convolutional Block Attention Module [15] with adaptive residual connection weights (ResCBAM) to enhance feature extraction through spatial-channel attention, focusing on defect regions while suppressing background interference. Subsequently, We apply Spatial and Channel Reconstruction Convolution (SCConv) [16] to eliminate feature redundancy in high-level representations. To improve discrimination in few-shot scenarios, Prototypical Networks (ProtoNet) strengthen feature separation between background and defects. Additionally, a cross-scale attention (CSA) mechanism enhances small defect detection through multi-scale feature integration. The CIoU loss [17] is adopted for precise defect localization during bounding box regression.

In our study, we performed comprehensive assessments using leather defect data sourced from actual leather manufacturing lines to verify the effectiveness of the proposed approach. To further evaluate the generalization capability of our model, we additionally introduced the publicly available NEU-DET dataset [18] for comprehensive evaluation.

In summary, this research endeavors to establish an efficient and precise automated detection method specifically designed to address the unique requirements and challenges associated with leather defect inspection. The proposed method seeks to enhance automation and efficiency in the quality inspection process of leather products, thereby contributing to the advancement of intelligent manufacturing within the leather industry.

2 Related Works

Deep learning has made remarkable advancements in the field of computer vision, leading to its increasing integration into industrial surface defect detection. One major breakthrough in this field is the application of CNNs, which effectively detect surface defects through their capacity to learn localized image features and hierarchical representations [19]. As research on CNN architectures has deepened, numerous CNN-based image detection networks have emerged, with widely used examples including Faster R-CNN [20], SSD [21], and the YOLO series [22].

2.1 Two-stage Detection

Faster R-CNN utilizes a two-stage architecture that integrates a Region Proposal Network to combine region proposal and object detection tasks. For example, Liu et al. [23] proposed an aircraft engine blade defect detection system based on improved Faster R-CNN that addresses the challenges of detecting tiny defects and discontinuous defects through RoI Align, Feature Pyramid Network, and improved non-maximum suppression algorithms. Similarly, Zhou et al. [24] combined Faster R-CNN with maximum entropy threshold segmentation, Canny edge detection, as well as projection feature and skeleton extraction methods to build a comprehensive crane surface crack detection and measurement system. Despite the widespread application of Faster R-CNN across various fields, its capacity for real-time processing often falls short of the rigorous demands associated with industrial surface detection.

2.2 One-stage Detection

To address the aforementioned issues, a growing number of scholars have transitioned to employing faster single-stage detection approaches, including SSD and YOLO. For example, Li et al. [25] introduced a lightweight detection model by integrating the SSD network with MobileNet architecture. Through a systematic optimization of network structure and parameter configuration, their approach achieved efficient real-time detection of surface defects on container sealing interfaces, including cracks and dents,

while preserving detection accuracy. Yan et al. [11] developed LGP-YOLO for light guide plate defect detection that uses specialized RFM and SDM modules to overcome challenges of complex backgrounds, low contrast, and multi-scale defects. Similarly, Pan et al. [26] proposed YOLO-ACF by integrating an Adaptive Complementary Fusion module into YOLOv5, improving photovoltaic panel defect detection by reducing false positives and undetected defects within intricate backgrounds. Additionally, to improve the detection of surface defects in hot-rolled strip steel, Huang et al. [27] incorporated the CSE module, Swin Transformer, and ASFF into YOLOv5, successfully overcoming difficulties in identifying small defects and distinguishing between defect classes. In addition to these methodologies, research has also explored the application of Generative Adversarial Networks (GANs) for defect detection, wherein adversarial training is employed to develop discriminators capable of identifying anomalies in images [28, 29].

3 Methods

3.1 Overall network architecture

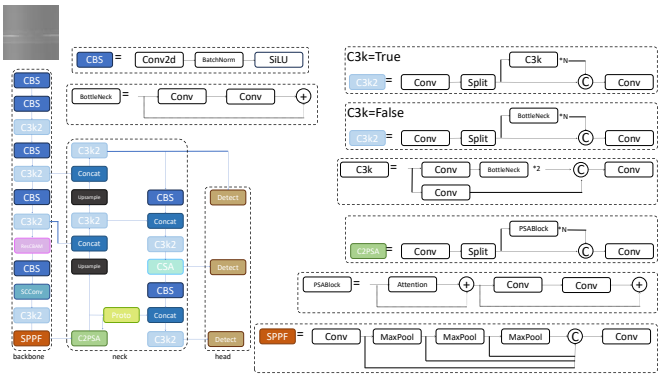

Fig. 2. DASC-YOLO network architecture

A schematic diagram of the proposed model architecture is presented in **Fig. 2**. The model foundation employs YOLOv11, the latest version of the YOLO series. While preserving the original head structure of YOLOv11, modifications have been made to its backbone and neck components to improve the detection of surface defects in leather materials, all while ensuring optimal inference speed.

In the backbone network, the ResCBAM and SCConv modules are chosen to reduce feature redundancy and background interference: ResCBAM is preferred over the standard CBAM due to its residual connection that preserves critical spatial information while refining attention-aware features, making it more effective for complex materials like leather; SCConv is selected over other convolution methods because of its dual-reconstruction mechanism, which effectively balances channel-wise feature compres-

sion and spatial enhancement, ensuring better preservation of high-level semantic features without losing spatial detail. For the neck, The CSA module improves multi-scale feature fusion by adopting a simplified, RFBNet-inspired design that reduces feature interaction complexity. By utilizing fewer branches, it enhances computational efficiency while preserving high accuracy in defect detection across multiple scales. And ProtoNet enhances model generalization under data scarcity by leveraging prototype learning. By constructing a learnable defect prototype representation space, ProtoNet enforces feature compactness within classes while maintaining separation between different classes, allowing the model to effectively recognize various defect types even with limited samples, thereby reducing overfitting and improving performance on unseen data. This integrated framework achieves balanced accuracy and efficiency for industrial defect detection with limited samples.

3.2 Cross-Scale Attention

To overcome the difficulties arising from substantial variations in defect dimensions, inspired by the RFBNet framework [30], we have designed and incorporated a Cross-Scale Attention (CSA) module into the neck of YOLOv11. This module significantly improves the perception of defects across multiple scales by employing a cross-scale feature interaction mechanism, while satisfying real-time requirements for industrial surface detection. The core architecture comprises three essential components: a small-object enhancement branch, a global context branch, and an adaptive fusion unit.

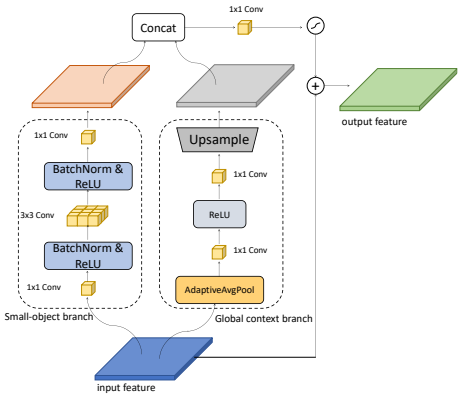

Fig. 3. Schematic diagram of the Cross-Scale Attention

As shown in **Fig. 3**, the CSA module is designed with a dual-branch architecture. The small-object enhancement branch extracts local detail features through a compression-expansion convolutional layer, effectively reducing computational costs while preserving microscopic defect textures. The global context branch is tasked with capturing semantic information via adaptive average pooling. It processes features through a convolutional layer with channel compression ratio of 4 to generate channel attention weights, and aligns spatial dimensions using bilinear interpolation.

The features derived from both the small-object enhancement branch and the global context branch are merged using learnable weighting, followed by feature fusion using 1×1 convolutions, and the attention scores will be subsequently computed. The computational formulation of the CSA module is expressed as follows:

$$F_{fusion} = Concat[\lambda_s \cdot F_s(x),\ \lambda_g \cdot F_g(x)] \tag{1}$$

$$F_{csa} = \alpha \cdot Sigmoid\left(W_{fusion} \cdot F_{fusion}\right) \cdot x + (1 - \alpha) \cdot x \tag{2}$$

In this formulation, x represents the input feature map. $F_s(x)$ and $F_g(x)$ denote the outputs from the small-object branch and global context branch respectively, with λ_s and λ_g being their corresponding learnable weights. Following concatenation of these outputs, a 1×1 convolution is employed for feature fusion, after which the attention scores are computed through a sigmoid activation function. The resultant features are then integrated with the original input features via residual connection, where α serves as an additional learnable weight parameter governing this integration.

3.3 ProtoNet

To address the challenge of constrained model accuracy resulting from the limited leather defect samples, as well as the variability in the size and morphology of these defects, we have proposed a prototype learning-based ProtoNet module. This module directs the network to achieve feature space compactness within classes and separation between classes through the creation of a learnable defect prototype space. As depicted in **Fig. 4**, the ProtoNet consists of three key components: learnable prototypes, similarity computation, and adaptive residual connection.

We have designed a lightweight feature encoder aimed at reducing the computational overhead from prototype matching operations. The architecture employs 1×1 convolutional layers for dimensionality reduction followed by group convolution (groups=4) to achieve channel decoupling, complemented by a single Dropout layer (rate=0.1) to prevent overfitting in few-sample scenarios. The encoded features are compressed into compact representations through adaptive pooling and flattening operations, ensuring efficient prototype similarity computation while maintaining discriminative power.

In the context of prototype representation learning, the module initialization phase constructs a library of learnable prototypes $\mathbf{P} \in R^{K \times C}$, where K is the predefined number of prototypes. Then, L2 normalization is then applied to constrain the vector space of the prototype vectors $\mathbf{p}_k$ and the encoded features $\mathbf{f}_i$. The pixel-wise similarity score matrix:

$$\mathbf{S} = \frac{\mathbf{f}_i \mathbf{p}_k^T}{\tau} \tag{3}$$

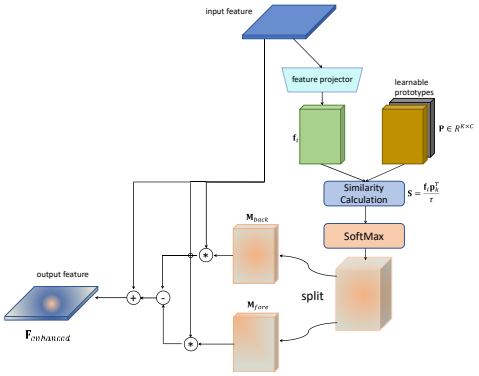

Fig. 4. Schematic diagram of the ProtoNet

Where τ is a learnable temperature coefficient that adjusts the sharpness of the matching probability distribution. When generating spatial attention maps via Softmax, a decoupled branch strategy is adopted: the background prototype produces a single-channel suppression mask, while multiple defect prototypes generate enhancement masks through max-pooling, effectively decoupling normal texture regions from potential defect responses. At the final stage, an adaptive residual weighting mechanism is employed to augment the input features through prototype attention.

$$\mathbf{F}_{enhanced} = \mathbf{F}_{input} \odot (1 + \alpha \cdot \mathbf{M}_{fore}) - \mathbf{F}_{input} \odot (\beta \cdot \mathbf{M}_{back}) \tag{4}$$

Here, α and β are learnable parameters initialized to 0.5, while $\mathbf{M}_{fore}$ and $\mathbf{M}_{back}$ represent the foreground and background attention maps respectively. The parameter α dynamically modulates the feature enhancement intensity in defective regions, whereas β suppresses feature interference from background areas. The residual structure preserves the integrity of original features, thereby avoiding potential feature deviation caused by excessive reliance on prototype matching.

3.4 Enhanced Backbone Design and Loss function

To enhance the backbone network of YOLOv11, modifications have been made to maintain the efficiency of feature extraction while reducing the interference caused by intricate background textures present in leather materials. We propose two key improvements: Firstly, we have restructured the standard CBAM attention mechanism by incorporating residual connections, creating a ResCBAM module that implements a dual-path "Identity Mapping + Attention Refinement" architecture through element-wise summation of original and CBAM-processed features. This residual-enhanced design preserves critical spatial information while refining attention-aware features. The ResCBAM structure is depicted in **Fig. 5** (a). Secondly, we have incorporated the standardized SCConv module into deeper network layers to address feature redundancy and background interference challenges. The dual-reconstruction mechanism of SCConv

strategically balances channel-wise feature compression and spatial enhancement, particularly crucial for preserving semantic richness in high-level features that often suffer from spatial detail degradation. Collectively, these synergistic modifications maintain computational efficiency while improving feature discrimination capability for complex surface defect patterns.

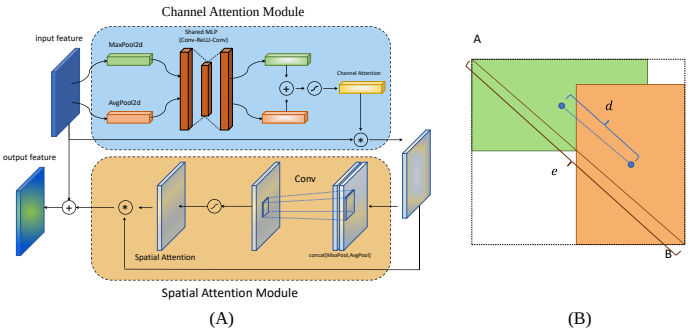

Fig. 5. Schematic diagram of (A) ResCBAM and (B) CIoU

For the defect detection tasks, the accurate regression of bounding boxes is essential for effective defect localization. Traditional IoU loss functions, limited by their sole focus on overlap area, often result in suboptimal localization. To address this, we use the CIoU loss, which enhances geometric optimization by integrating overlap metrics, center distance penalties, and aspect ratio constraints, enabling multidimensional refinement of bounding box accuracy [17]. IoU can be mathematically expressed as:

$$IoU = \frac{|A \cap B|}{|A \cup B|} \tag{5}$$

And the CIoU Loss is defined as:

$$\mathcal{L} = 1 - \mathrm{IoU} + \frac{d}{e^2} + \alpha v \tag{6}$$

As illustrated in the **Fig. 5** (b), d indicates the distance between the centroids of the predicted bounding box and the ground truth box, while e denotes the diagonal distance of their minimal enclosing rectangle. The definitions of α and v are as follows.

$$\alpha = \frac{v}{(1-\mathrm{IoU})+v} \tag{7}$$

$$v = \frac{4}{\pi^2}\left(\arctan\left(\frac{w^{gt}}{h^{gt}}\right) - \arctan\left(\frac{w}{h}\right)\right)^2 \tag{8}$$

Here, (w^{gt}, h^{gt}) and (w, h) denote the width and height of the ground truth bounding box and the predicted bounding box, respectively.

4 Experiments

4.1 Dataset

High-resolution images were acquired utilizing 8K line scanning cameras on actual leather production lines. These images were subsequently cropped to dimensions of 2048×2048 pixels to align with practical inspection standards. Expert quality inspectors annotated defect types including black and white dots, color streaks, and concatenation flaws. To validate that the proposed method in this study extends beyond leather applications, comparative experiments were also conducted using the NEU-DET dataset [18] which comprises 1,800 grayscale images (200×200 pixels) depicting steel strip defects categorized into six distinct types. Each defect category, such as scratches and inclusions, is represented by 300 samples.

To ensure robust validation, a 5-fold cross-validation protocol was applied to the training set, following an 8:2 split of the leather defect and NEU-DET datasets into training and test partitions. Strict separation between all subsets guaranteed no image overlap, thereby upholding evaluation reliability. The methodology for dataset partitioning is detailed in **Table 1**.

Table 1. Details of Leather Defect Dataset and NEU-DET

Dataset	Defect Type	Train	Test	Total
Leather Defect Dataset	Black dot	219	55	274
	White dot	81	21	102
	Concat	142	36	178
	Colorstreak	95	24	119
Total		537	136	673
NEU-DET	crazing	240	60	300
	inclusion	240	60	300
	patches	240	60	300
	pitted	240	60	300
	Rolled-in	240	60	300
	scratch	240	60	300
Total		1440	360	1800

4.2 Experiments setup

The experiments were performed utilizing a high-performance computing system featuring an Intel Core i7-7820X CPU operating at 3.60GHz, complemented by 80GB of RAM. The computational capabilities were further enhanced by the inclusion of dual NVIDIA GeForce RTX 3090 GPUs, each possessing 24GB of dedicated memory. Ubuntu 20.04 LTS served as the foundational operating system for the software environ-

ment, which included support for CUDA 12.2 acceleration. Program execution was facilitated through Python version 3.10.0, while PyTorch version 2.5.1 served as the primary framework for deep learning applications.

Several essential hyperparameters are configured in the training process to maintain a balance between computational cost and model performance. For basic settings, high-resolution 2048×2048 images are processed with a batch size of 16 across 300 training epochs. We employ the AdamW optimizer with an initial learning rate of 0.001, which decays to 5% of its starting value. Overfitting is addressed through the application of L2 regularization, configured with a 0.0005 weight decay parameter.

In the context of leather defect detection within industrial applications, the lightest YOLO configuration serves as the backbone network, strategically chosen to preserve both detection accuracy and processing speed. All comparative models are configured to maintain equivalent minimal-scale parameters. In public benchmark assessments that emphasize accuracy, our framework, along with YOLO-based comparative models, consistently employs large-scale architectures to facilitate equitable performance comparisons.

4.3 Metrics

To ensure accurate and effective evaluation of model performance in our experimental results, we selected the following objective evaluation metrics:

Precision reflects prediction accuracy, while recall indicates detection completeness. In object detection, high precision with low recall suggests under-detection from over-caution, whereas low precision with high recall signals over-detection due to leniency. The definitions of Precision and Recall are as follows:

$$Precision = \frac{\mathrm{TP}}{\mathrm{TP+FP}},\ Recall = \frac{\mathrm{TP}}{\mathrm{TP+FN}} \tag{9}$$

For a thorough evaluation of the model's effectiveness, we adopted the F1-score—a balanced measure combining precision and recall. This metric balances false positives and negatives while offering a robust performance measure for class-imbalanced scenarios. The definitions of F1-score is as follows:

$$F1 = \frac{2}{\frac{1}{\mathrm{Precision}}+\frac{1}{\mathrm{Recall}}} = \frac{2\times\mathrm{Precision}\times\mathrm{Recall}}{\mathrm{Precision+Recall}} \tag{10}$$

AP (Average Precision) and mAP (mean Average Precision) are standard evaluation metrics in object detection. AP comprehensively evaluates single-class detection performance, while mAP extends this to multi-class scenarios by averaging AP values across all categories, enabling unbiased model comparisons through a unified benchmark. The definitions of AP and mAP are as follows:

$$AP = \int_0^1 P(R)\mathrm{d}R,\ mAP = \frac{\sum_1^N \int_0^1 P(R)\mathrm{d}R}{N} \tag{11}$$

4.4 Comparison experiment on leather dataset

To verify our model's efficacy in leather defect detection against existing methods, all models were trained from scratch under identical experimental conditions with optimized parameters. We selected representative YOLO mdoels (v5, v8, and v11) as industry-standard benchmarks.

For fair comparison, our proposed model and the YOLOv5/YOLOv8 implementations all employed their smallest available variants, while YOLOv11 was evaluated using models ranging from N to L sizes during training and testing. The experimental results are presented in **Table 2**.

Table 2. Comparison of different object detection models on leather defect dataset

Method	AP(%)				mAP	Params	GFlops	FPS
	Black	White	Color	Concat				
Faster R-CNN	75.6	70.4	72.7	63.8	70.6	165.69M	76.80	67.7
SSD	74.7	66.9	73.4	70.1	71.3	26.28M	1443.28	8.43
Yolov5n	**83.4**	68.8	78.3	76.1	76.7	4.13M	37.30	94.2
Yolov8n	74.4	43.5	57.6	56.2	57.9	3.16M	45.35	92.0
Yolo11n	82.8	68.3	**92.3**	65.4	77.2	2.59M	32.98	87.0.
Yolo11s	78.2	72.5	84.7	73.6	77.6	9.43M	110.33	44.0
Yolo11l	73.7	70.7	88.3	79.1	78.0	25.31M	446.84	14.9
ours	78.3	**77.3**	86.9	**81.6**	**81.0**	3.02M	35.80	70.3

As shown in the **Table 2**, YOLOv5n and YOLO11n achieved the highest AP in black dot and color streak detection respectively, while our model demonstrated superior performance in white dot detection and concatenated tasks, attaining the highest AP values. Additionally, our model exhibited a higher mAP compared to other models, surpassing Faster R-CNN, SSD, YOLOv5n, YOLOv8n, YOLO11n, YOLO11s, and YOLO11l by 10.4%, 9.7%, 4.3%, 23.1%, 3.8%, 3.4%, and 3% respectively. Analyzing defect sample distribution and detection performance, our model better balances many-shot and few-shot accuracy. Based on the experimental outcomes, it is evident that both Faster R-CNN and SSD exhibit limitations in processing images with dimensions of 2048×2048, as evidenced by their relatively low mAP. Furthermore, SSD specifically suffers from a significant reduction in FPS when handling such oversized input images.

Our model adds only 0.43M parameters, the second most efficient design, with merely 2.82 GFLOPs increase over YOLOv11n's computational load. This optimization preserves real-time FPS while achieving superior detection performance compared to other YOLO variants and other detection models, demonstrating effective balance between accuracy and speed.

4.5 Ablation

To validate the significant role of our proposed modules and the modifications introduced for leather defect detection, we individually incorporated CBAM, ResCBAM,

CSA, ProtoNet, and SCConv into YOLO11 and conducted combination tests to investigate the contributions of different modules to the model performance.

The experimental results are summarized in **Table 3**. Integrating CBAM modules enhanced mAP and precision metrics, with residual-equipped ResCBAM achieving the strongest gains (1.3% mAP and 1.8% precision improvements over baseline). While recall slightly decreased, possibly due to attention over-focus compromising detection coverage. Although computational costs reduced FPS, the modules maintained parameter efficiency.

The CSA module achieved improvements of 1.2% mAP and 0.9% precision with minimal FPS reduction, demonstrating an effective accuracy-speed trade-off. While standalone SCConv and ProtoNet(K=2,4) integrations did not enhance mAP greatly, each contributed a 0.4% F1-score gain through precision/recall improvements. This indicates SCConv's spatial-channel decoupled convolutions mitigate redundant feature interference, while ProtoNet's prototype feature alignment enhances defect category discriminability. The ablation study results of our experiments on CBAM and SCConv are consistent with those reported by Zhang et al. [31] and Li et al. [32].

We also investigated the impact of varying prototype numbers (K) on ProtoNet. Experimental results demonstrate that ProtoNet achieves optimal performance when K=4, whereas configurations with K=2 or K=8 exhibit degraded results. This phenomenon may be attributed to insufficient prototype quantities failing to adequately match the data distribution (K=2) or excessive prototypes introducing noise that compromises model performance (K=8). Consequently, we adopted K=4 as the prototype configuration for final model.

Table 3. The result of ablation experiment on the leather defect dataset

Model	mAP	Pre	Recall	F1	Params	FPS
Yolo11	77.2	75.8	75.2	75.5	2.59M	87.1
Yolo11-CBAM	78.3	76.2	73.5	74.8	2.59M	80.8
Yolo11-ResCBAM	78.5	77.6	72.4	74.9	2.59M	79.9
Yolo11-CSA (w/o weighted fusion)	77.9	76.7	73.5	75.1	2.68M	83.9
Yolo11-CSA	78.4	76.7	74.3	75.5	2.68M	83.5
Yolo11-SCConv	77.5	76.7	75.0	75.8	2.71M	81.9
Yolo11-ProtoNet(K=2)	77.3	76.6	75.2	75.9	2.84M	80.5
Yolo11-ProtoNet(K=4)	78.0	76.5	75.4	75.9	2.84M	80.4
Yolo11-ProtoNet(K=8)	77.1	75.4	74.9	75.1	2.84M	79.7
Yolo11-ResCBAM-CSA	79.5	79.1	75.9	77.4	2.68M	78.1
Yolo11-ResCBAM-CSA-SCConv	80.4	79.8	76.2	78.0	2.81M	73.1
Ours	81.0	80.2	76.4	78.3	3.02M	70.3

Progressive combination of all four modules produced substantial improvements across all metrics: 3.8% mAP, 4.4% Precision, 1.2% Recall, and 2.8% F1-score. Comprehensive analysis reveals that our modified ResCBAM contributes most significantly

to mAP and Precision improvements, albeit with detectable speed compromise. Complementary modules achieve balanced F1-score enhancement through Precision and Recall optimization while maintaining reasonable detection speed, thus preventing performance imbalance from single-metric over-optimization.

To select an appropriate loss function, we evaluated multiple IoU-based losses including CIoU [17], EIoU [33], MDPIoU [34], PowerfulIoU [35], ShapeIoU [36], SIoU [37], and UnifiedIoU [38]. After extensive experimentation, CIoU was ultimately selected as the optimal choice owing to its stable loss convergence characteristics and superior mAP performance, with the comparative results visually presented in the **Fig. 6**.

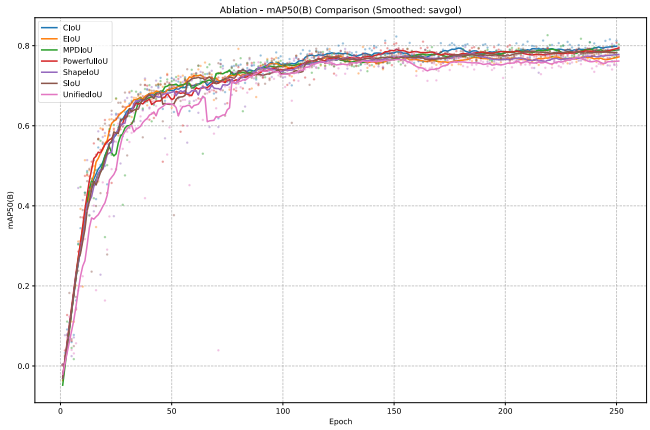

Fig. 6. IoU Loss Curve

4.6 Comparison experiment on NEU-DET dataset

For additional validation of the model's defect detection ability, we employed the NEU-DET benchmark dataset for training and testing. The performance metrics of comparative models were extracted from their original publications, while YOLO-series models were trained under identical software/hardware environments as our framework, employing optimized hyperparameters to ensure experimental fairness. For comprehensive performance evaluation, all YOLO-series models in this section adopted the large architecture to compare full-capacity models in steel surface defect detection. Within this dataset, crazing and rolled-in scale defects present particular identification challenges, attributable not only to low contrast with background textures but also to high inter-class similarity among defect categories - both representing critical issues in industrial surface defect inspection.

The experimental results are presented in the **Table 4**. MSFT-YOLO, LDD-YOLO, our method, improved Faster R-CNN, Efficient-D1, and SCRL-EMD-FR achieved the best detection performance on crazing, inclusion, patches, pitted, rolled-in, and scratches defects respectively. Our model maintains robust detection capabilities for low-contrast defects (crazing and rolled-in), while exhibiting superior performance on more visually

distinct defect types. Although YOLOv7 achieves the highest inference speed (104 FPS), our large-scale architecture strikes an optimal balance between detection accuracy and processing efficiency. Notably, our solution achieves state-of-the-art comprehensive performance with a mAP of 79.2%, indicating its overall superior performance in defect detection tasks.

Table 4. Comparison of different object detection models on NEU-DET dataset

Method	AP(%)						mAP(%)	FPS
	Cr	In	Pa	Pi	Ro	Sc		
Improved Yolo3[39]	38.9	73.7	93.5	74.8	60.7	91.4	72.2	64.5
Yolov5	33.0	84.1	90.4	80.5	60.4	84.5	72.2	72.3
Yolov6[14]	28.9	83.2	90.9	74.4	67.5	85.4	71.7	56.1
Yolov7[14]	47.4	83.2	93.7	84.9	55.4	88.7	75.6	104.0
Yolov8	36.1	83.6	89.3	84.4	66.2	93.2	75.5	77.4
Yolo11	49.8	76.7	93.7	85.2	64.3	84.2	75.6	72.9
Improved Yolox[40]	55.1	83.0	93.6	86.1	59.7	84.2	77.0	100
SSD300[39]	41.1	79.6	83.9	83.9	62.1	83.6	71.4	37.4
SSD500[39]	41.7	76.3	86.3	85.1	58.1	85.6	72.4	29.0
EfficientDet-D0[41]	56.8	69.8	88.2	78.4	69.3	25.9	64.7	/
EfficientDet-D1[41]	49.4	77.5	88.7	81.3	**72.7**	49.3	68.8	/
Faster R-CNN[42]	50.1	79.1	79.2	**87.4**	64.9	90.5	75.1	/
Libra R-CNN[43]	38.3	81.3	88.4	81.3	67.8	88.2	73.3	12.0
RT-DETR[44]	37.2	78.5	91.2	79.0	59.9	90.4	72.6	30.1
SCRL-EMD-FR[45]	46.8	83.0	92.8	85.2	58.5	**95.1**	76.9	14.4
SCRL-EMD-RN[45]	45.3	73.4	93.4	83.5	56.1	84.3	72.7	16.1
HA-YOLO[46]	52.4	80.4	86.9	75.0	66.1	89.8	75.1	/
ES-NET[47]	56.0	87.6	88.3	87.4	60.4	94.9	79.1	53.0
SN-TOOD[48]	39.2	79.2	86.6	80.5	56.5	85.2	71.2	/
MSFT-YOLO[49]	**56.9**	80.8	93.5	82.1	52.7	83.5	74.9	30.6
LDD-YOLO[50]	42.2	**87.8**	94.1	86.3	65.6	93.4	78.2	/
Ours	55.7	79.6	**94.9**	84.9	68.5	91.8	**79.2**	55.8

5 Conclusion

This study introduces an advanced YOLOv11 framework aimed at addressing critical challenges associated with the detection of surface defects in industrial environments. Specifically, it focuses on issues related to multi-scale defect variations, a scarcity of defect samples, and complex texture interference. The research delineates three key innovations: 1) An improved backbone network that integrates ResCBAM and SCConv modules, which effectively mitigates background noise from intricate leather textures while maintaining efficient feature extraction and enhancing the representation of discriminative defect features; 2) A cross-scale attention mechanism that utilizes feature fusion strategies to address scale sensitivity challenges across defects of varying sizes;

and 3) A ProtoNet employing CIoU loss functions that facilitates the synergistic optimization of precise localization and classification in scenarios with limited sample availability, thereby enhancing adaptability to the morphological diversity of leather defects. The model has been rigorously evaluated using both self-collected datasets of leather defects and established public benchmarks for steel surface defects, demonstrating superior performance, computational efficiency, and real-time inference capabilities. Future research endeavors will focus on developing adaptive optimization strategies tailored to diverse industrial inspection contexts, thereby accommodating varying requirements for speed and accuracy.

Acknowledgments. This work was supported by General Program of Chongqing Natural Science Foundation [CSTB2024NSCQ-MSX0729]; Strategic cooperation project between Chongqing and Chinese Academy of Agricultural Sciences [12].

Disclosure of Interests. The authors have no competing interests to declare that are relevant to the content of this article.

References

1. Aslam M, Khan TM, Naqvi SS, et al.: On the Application of Automated Machine Vision for Leather Defect Inspection and Grading: A Survey. IEEE Access 7:176065–176086 (2019)
2. Campbell LE, Connor ,Robert J., Whitehead ,Julie M., and Washer GA.: Human factors affecting visual inspection of fatigue cracking in steel bridges. Struct Infrastruct Eng 17:1447–1458 (2021)
3. Lee C-F, Chen Y-C, Shen J-J, Rehman AU.: Lightweight Leather Surface Defect Inspection Model Design for Fast Classification and Segmentation. Symmetry 17:358 (2025)
4. Demir K, Ay M, Cavas M, Demir F.: Automated steel surface defect detection and classification using a new deep learning-based approach. Neural Comput Appl 35:8389–8406. https://doi.org/10.1007/s00521-022-08112-5 (2023)
5. Natarajan V, Hung T-Y, Vaikundam S, Chia L-T.: Convolutional networks for voting-based anomaly classification in metal surface inspection. In: 2017 IEEE International Conference on Industrial Technology (ICIT). pp 986–991 (2017)
6. Fang B, Long X, Sun F, et al.: Tactile-Based Fabric Defect Detection Using Convolutional Neural Network With Attention Mechanism. IEEE Trans Instrum Meas 71:1–9 (2022)
7. Qiu L, Wu X, Yu Z .: A High-Efficiency Fully Convolutional Networks for Pixel-Wise Surface Defect Detection. IEEE Access 7:15884–15893 (2019)
8. Khanam R, Hussain M, Hill R, Allen P.: A Comprehensive Review of Convolutional Neural Networks for Defect Detection in Industrial Applications. IEEE Access 12:94250–94295. (2024)
9. Parag Kohli PK.: Leather Quality Estimation Using an Automated Machine Vision System. IOSR J Electron Commun Eng 6:44–47 (2013)
10. Chen S-Y, Cheng Y-C, Yang W-L, Wang M-Y.: Surface Defect Detection of Wet-Blue Leather Using Hyperspectral Imaging. IEEE Access 9:127685–127702 (2021)
11. Wan Y, Li J.: LGP-YOLO: an efficient convolutional neural network for surface defect detection of light guide plate. Complex Intell Syst 10:2083–2105 (2024)
12. Liu Y, Zhang C, Dong X.: A survey of real-time surface defect inspection methods based on deep learning. Artif Intell Rev 56:12131–12170 (2023)

13. Hu X, Lin S.: DFFNet: a lightweight approach for efficient feature-optimized fusion in steel strip surface defect detection. Complex Intell Syst 10:6705–6723 (2024)
14. Liu S, Li J.: EC-PFN: a multiscale woven fusion network for industrial product surface defect detection. Complex Intell Syst 11:59 (2024)
15. Woo S, Park J, Lee J-Y, Kweon IS.: CBAM: Convolutional Block Attention Module. In: Ferrari V, Hebert M, Sminchisescu C, Weiss Y (eds) Computer Vision – ECCV 2018. Springer International Publishing, Cham, pp 3–19 (2018)
16. Li J, Wen Y, He L.: SCConv: Spatial and Channel Reconstruction Convolution for Feature Redundancy. In: 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp 6153–6162 (2023)
17. Zheng Z, Wang P, Ren D, et al.: Enhancing Geometric Factors in Model Learning and Inference for Object Detection and Instance Segmentation. IEEE Trans Cybern 52:8574–8586 (2022)
18. 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 69:1493–1504 (2020)
19. Li Z, Liu F, Yang W, et al.: A Survey of Convolutional Neural Networks: Analysis, Applications, and Prospects. IEEE Trans Neural Netw Learn Syst 33:6999–7019 (2022)
20. 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 39:1137–1149 (2017)
21. Liu W, Anguelov D, Erhan D, et al.: SSD: Single Shot MultiBox Detector. In: Leibe B, Matas J, Sebe N, Welling M (eds) Computer Vision – ECCV 2016. Springer International Publishing, Cham, pp 21–37 (2016)
22. Redmon J, Divvala S, Girshick R, Farhadi A.: You Only Look Once: Unified, Real-Time Object Detection. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp 779–788 (2016)
23. Yixuan L, Dongbo W, Jiawei L, Hui W.: Aeroengine Blade Surface Defect Detection System Based on Improved Faster RCNN. Int J Intell Syst 2023:1992415 (2023)
24. Zhou Q, Ding S, Qing G, Hu J.: UAV vision detection method for crane surface cracks based on Faster R-CNN and image segmentation. J Civ Struct Health Monit 12:845–855 (2022)
25. Li Y, Huang H, Xie Q, et al.: Research on a Surface Defect Detection Algorithm Based on MobileNet-SSD. Appl Sci 8:1678 (2018)
26. Pan W, Sun X, Wang Y, et al.: Enhanced photovoltaic panel defect detection via adaptive complementary fusion in YOLO-ACF. Sci Rep 14:26425 (2024)
27. Huang X, Zhu J, Huo Y.: SSA-YOLO: An Improved YOLO for Hot-Rolled Strip Steel Surface Defect Detection. IEEE Trans Instrum Meas 73:1–17 (2024)
28. Guo Y, Zhong L, Qiu Y, et al.: Using ISU-GAN for unsupervised small sample defect detection. Sci Rep 12:11604 (2022)
29. Liu W, Wang C, Zhang Y.: Industrial surface defect detection by multi-scale Inpainting-GAN. Vis Comput (2024)
30. Liu S, Huang D, Wang Y.: Receptive Field Block Net for Accurate and Fast Object Detection. In: Ferrari V, Hebert M, Sminchisescu C, Weiss Y (eds) Computer Vision – ECCV 2018. Springer International Publishing, Cham, pp 404–419 (2018)
31. Zhang H, Li S, Miao Q, et al.: Surface defect detection of hot rolled steel based on multi-scale feature fusion and attention mechanism residual block. Sci Rep 14:7671 (2024)
32. Li H, Yi Z, Mei L, et al (2024) SCFNet: Lightweight Steel Defect Detection Network Based on Spatial Channel Reorganization and Weighted Jump Fusion. Processes 12:931 (2024)
33. Zhang Y-F, Ren W, Zhang Z, et al (2022) Focal and efficient IOU loss for accurate bounding box regression. Neurocomputing 506:146–157 (2024)

34. Ma S, Xu Y.: MPDIoU: A Loss for Efficient and Accurate Bounding Box Regression. In: arXiv preprint arXiv:2307.07662 (2023)
35. Liu C, Wang K, Li Q, et al.: Powerful-IoU: More straightforward and faster bounding box regression loss with a nonmonotonic focusing mechanism. Neural Netw 170:276–284 (2024)
36. Zhang H, Zhang S.: Shape-IoU: More Accurate Metric considering Bounding Box Shape and Scale. In: arXiv preprint arXiv:2312.17663 (2023)
37. Gevorgyan Z.: SIoU Loss: More Powerful Learning for Bounding Box Regression. In: arXiv preprint arXiv:2205.12740 (2022)
38. Luo X, Cai Z, Shao B, Wang Y.: Unified-IoU: For High-Quality Object Detection. In: arXiv preprint arXiv:2408.06636 (2024).
39. Kou X, Liu S, Cheng K, Qian Y.: Development of a YOLO-V3-based model for detecting defects on steel strip surface. Measurement 182:109454 (2021)
40. Li C, Xu A, Cai Y, Zhang Q.: Improved YOLOX-based Method for Steel Surface Defect Detection. In: 2023 5th International Academic Exchange Conference on Science and Technology Innovation (IAECST). pp 929–934 (2023)
41. Zhang Y, Wang W, Li Z, et al.: Development of a cross-scale weighted feature fusion network for hot-rolled steel surface defect detection. Eng Appl Artif Intell 117:105628 (2023)
42. Zhao W, Chen F, Huang H, et al.: A New Steel Defect Detection Algorithm Based on Deep Learning. Comput Intell Neurosci 2021:5592878 (2021)
43. Pang J, Chen K, Shi J, et al.: Libra R-CNN: Towards Balanced Learning for Object Detection. pp 821–830 (2019)
44. Zhao Y, Lv W, Xu S, et al.: DETRs Beat YOLOs on Real-time Object Detection. In: 2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp 16965–16974 (2024)
45. Hu X, Yang J, Jiang F, et al.: Steel surface defect detection based on self-supervised contrastive representation learning with matching metric. Appl Soft Comput 145:110578 (2023)
46. Wang Y, Duan J, Huang J, Fu K.: HA-YOLO: A Real-Time Method for Detecting Surface Defects in Steel. In: 2024 5th International Conference on Artificial Intelligence and Computer Engineering (ICAICE). pp 1038–1041 (2024)
47. Yu X, Lyu W, Zhou D, et al.: ES-Net: Efficient Scale-Aware Network for Tiny Defect Detection. IEEE Trans Instrum Meas 71:1–14 (2022)
48. Liu G, Chu M, Gong R.: SN-TOOD: An Improved TOOD-Based Steel Surface Defect Detection. In: 2024 International Annual Conference on Complex Systems and Intelligent Science (CSIS-IAC). pp 413–418 (2024)
49. Guo Z, Wang C, Yang G, et al.: MSFT-YOLO: Improved YOLOv5 Based on Transformer for Detecting Defects of Steel Surface. Sensors 22:3467 (2022)
50. Zhang Y, Li A, Li Z, et al.: LDD-YOLO: An Improved Lightweight Detection Method for Steel Surface Defects Based on YOLOv8. In: 2024 IEEE International Conference on Systems, Man, and Cybernetics (SMC). pp 7–13 (2024)