SIG-Net: Object Detection Networks Focusing on Minor Defects in Complex Contexts

Xin Wang, Xue Li, Ziyang Li, Jiong Yu, Wenjing Li
School of Software Engineering, Xinjiang University
Urumqi, China
Email: 107552304985@stu.xju.edu.cn, {lixue, liziyang, yujiong, liwenjing}@xju.edu.cn

***Abstract*—Automated detection of steel surface defects is vital for ensuring product quality and safety. In complex textured backgrounds, defects are typically minute and low in contrast, making them easily obscured or confused with surrounding patterns and leading to frequent missed and false detections. To address this challenge, we propose SIG-Net, a lightweight detection network tailored for small defects in complex contexts. The framework integrates three core modules: the Small Object-Aware Refined Feature Pyramid (SOAR-FPN), which injects high-resolution details into P3 via SPDConv and enhances representation through CSP-OmniKernel fusion; the Inter-Layer Sparse Guidance Module (ISGM), which applies cross-scale sparse attention where high-level semantics guide low-level features to suppress noise and highlight defects; and the Global Edge Information Transfer (GEIT), which generates multi-scale edge features from shallow layers and propagates them across the network to improve boundary perception. Experimental results on the NEU-DET and GC10-DET datasets demonstrate that SIG-Net achieves mAP_{50} values of 82.2% and 83.4%, respectively, while maintaining a lightweight architecture, significantly outperforming existing methods and effectively addressing the challenge of small defect detection under complex backgrounds.**

***Index Terms*—Defect detection, Deep learning, Feature fusion, Attention mechanism.**

I. Introduction

As a key material in construction and infrastructure, the surface quality of steel directly affects product performance and safety [1]. Common defects such as cracks, scratches, and pits not only reduce quality but may also pose serious risks [2]. Automated detection is thus essential for safety and cost control. However, defects are often minute, low in contrast, irregular in shape, and easily obscured by complex textures, leading to frequent missed or false detections. Complex backgrounds further introduce high-frequency interference that weakens edge and feature responses, making representation and detection extremely challenging [3]. Traditional methods (e.g., manual inspection or non-destructive testing) are inefficient, costly, and unreliable [4]. Recently, deep learning, particularly CNN-based single-stage detectors such as YOLO [5], has shown great promise for defect detection. Yet in scenarios with ultra-small defects and background similarity, foreground-background confusion and feature loss remain major obstacles for practical deployment. In response, a series of task-optimized detectors have been proposed. WSS-YOLO [6] introduces WIoU dynamic focusing and C2f-DSC for adaptive receptive fields; Zhang et al.'s [7] HDFA-Net decouples high- and low-frequency features to improve small defect characterization; BiFPN [8] employs learnable weights for bidirectional fusion; FocusDet [9] supplements bottom-up shallow paths to compensate for downsampling losses; and MSAF-YOLOv8n [10] integrates multi-scale fusion with lightweight convolutions to enhance robustness. While effective, these methods remain limited in ultra-small defect scenarios due to missing fine-grained features, background interference, and insufficient edge perception, highlighting the need for novel architectures. Therefore, this paper proposes SIG-Net, a small imperfection detection network specifically designed for complex backgrounds. The network enhances detection performance through three core modules: firstly, SOAR-FPN employs spatial downsampling convolutions (SPDConv [11]) to extract high-resolution details and inject them into layer P3, while the CSP-OmniKernel module strengthens feature fusion capabilities for subtle imperfections. Secondly, ISGM employs a coarse-to-fine cross-scale sparse attention mechanism, where higher-level features guide lower-level processing to highlight potential defects while suppressing background noise. Thirdly, GEIT generates multi-scale edge maps from shallow layers and propagates them to the backbone network, achieving global edge fusion to enhance boundary perception and localisation stability.

II. Prepare Your Paper Before Styling

we propose an enhanced detection network, SIG-Net, based on YOLOv11-n for industrial surface defect detection. The proposed SIG-Net architecture, illustrated in Fig.1, comprises a backbone network with neck and head structures. Addressing the challenges outlined, this chapter introduces three innovative modules: SOAR-FPN, the ISGM, and GEIT.

A. Design of a Small-Object Aware Refined Feature Pyramid Network

Minor defect objects are often diminutive in scale, rendering them susceptible to gradual loss of detail within the high-level features of deep neural networks. Conventional detection layers such as P3, P4, and P5 struggle to adequately capture these nuances. Particularly in scenarios with complex background noise, defect features become obscured or diminished, resulting in elevated rates of detection failure. Traditional approaches introduce a P2 layer output at the detection head to leverage higher-resolution features for small target detection.

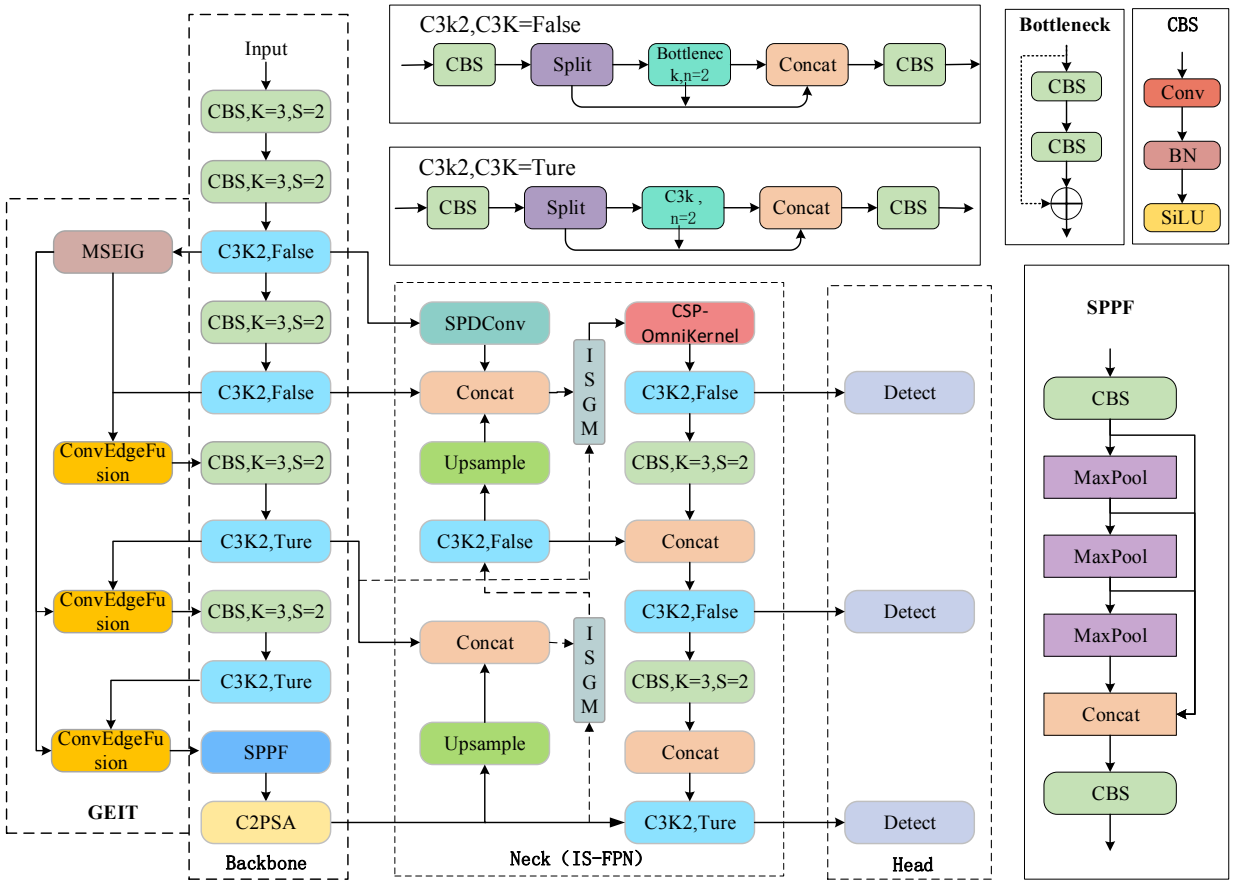

Fig. 1. Overall framework of the proposed SIG-Net.

However, directly adding a P2 detection layer substantially increases computational complexity and post-processing overhead.

To address this, this paper proposes SOAR-FPN (Fig.2) based on the PAFPN architecture from the YOLO series. Its structure comprises an SPDConv downsampling module and a CSP-OmniKernel feature fusion module, enhancing small object feature representation without requiring an additional P2 detection layer.

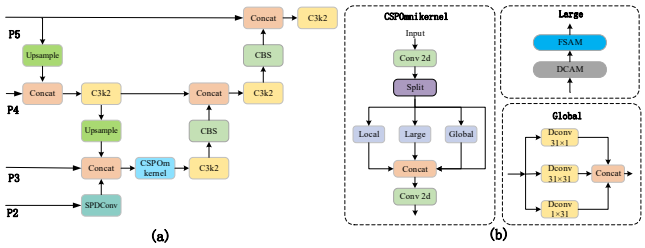

Fig. 2. (a) is the SOAR-FPN overall architecture. (b) is the structure of each module.

Firstly, regarding small object feature injection, we do not augment the P2 output but instead employ SPDConv to inject P2's high-resolution details into P3. SPDConv partitions P2 into 2×2 sub-patches via space-to-depth conversion, concatenates them along the channel dimension, and then fuses them with P3 through convolution. This approach preserves detail enhancement while avoiding the computational and post-processing overhead associated with introducing new detection heads.Secondly, in terms of feature fusion, we designed the CSP-OmniKernel. The input channels are proportionally divided into an identity pass-through branch and an OKM branch, with the OKM branch comprising qC channels and the identity branch comprising $(1-q)C$ channels. Optimal performance is achieved when q=0.4.

Feature map M1 undergoes processing through the OKM to produce feature map M3. Finally, feature maps M2 and M3 from the identity branch are concatenated, and feature fusion is performed to yield feature map M4.

$$M_3 = \mathrm{OKM}(M_1), M_4 = \mathrm{Conv2d}(\mathrm{Concat}(M_2, M_3)) \quad (1)$$

Through this design, we have effectively embedded the OKM into the CSP architecture, forming the CSP-OmniKernel module. Compared to standard FPN that merely fuse features sequentially layer by layer, our multi-branch fusion enables customised feature extraction tailored for small objects, significantly enhancing the richness of feature representation for subtle defects.

B. Design of Inter-layer Sparse Guided Module

Against complex backgrounds, low-level details and high-level semantics frequently misalign, causing background features to intrude and weaken defect object representations. Traditional Feature Pyramid Networks (FPNs) merely fuse features layer-by-layer, lacking selective focus on critical regions. While common attention or Transformer architectures can weight features, they rely on homogeneous QKV, hindering effective utilisation of cross-layer complementary information.This paper proposes the Inter-Scale Sparsity Guiding Module (ISGM), as illustrated in Fig.3.Our innovation lies in the observation that conventional attention mechanisms and Transformer architectures compute similarity scores using homogeneous (Q, K, V) representations, whereas the proposed ISGM adopts heterogeneous (Q, K, V). Moreover, building upon the SOAR-FPN, we construct the IS-FPN module by taking feature maps from adjacent scales (e.g., P3 and P4) as inputs, as depicted in the Neck section of Fig. 1, thereby enabling explicit guidance from high-level semantic features to low-level detailed representations. In addition, a two-stage attention mechanism is introduced: coarse-grained attention is first employed to filter potential defect regions, followed by sparse fine-grained attention at critical locations, achieving an effective combination of global guidance and local enhancement.

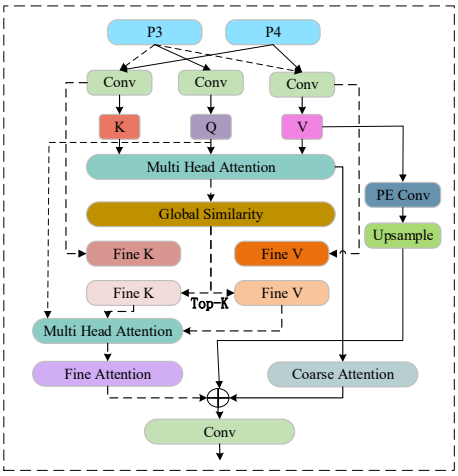

Fig. 3. ISGM Overall Architecture.

First, ISGM employs high-level features to provide regional guidance for low-level features. Let X denote the low-level feature map and U denote the adjacent high-level feature map. Initially, a cross-layer QKV is generated through convolutional mapping.Coarse-grained cross-layer attention enables high-level semantics to guide low-level regions, as shown in Eq.(3).

$$Q = \mathrm{Conv}(X), \quad K = \mathrm{Conv}(U), \quad V = \mathrm{Conv}(U). \tag{2}$$

$$O_{\mathrm{coarse}} = \mathrm{Softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{3}$$

Subsequently, a set Ω of positions exhibiting the highest responses is selected from the coarse-grained attention. These

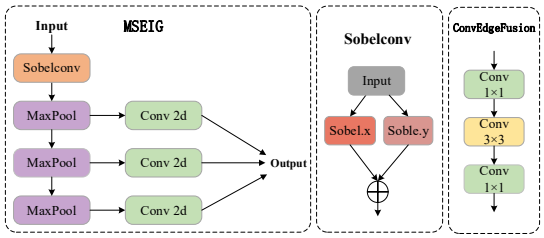

Fig. 4. MSEIG and ConvEdgeFusion Overall Architecture.

high-response locations correspond to regions on the original fine-grained feature map most likely to contain defect details. Fine-grained attention is then applied to these critical areas. Through sparse fine-grained attention, the network selectively enhances the feature representations of these candidate defect regions, boosting texture contrast along crack edges and amplifying response values for scratch pixels. Finally, the results from both stages are concatenated and convolutionally fused.

$$F_{fine} = \mathrm{Softmax}\left(\frac{Q_\Omega K_\Omega^T}{\sqrt{\mathrm{d}}}\right)V_\Omega. \tag{4}$$

Through cross-layer non-homogeneous QKV and two-stage sparse attention, ISGM selectively enhances low-level details under high-level semantic guidance. This significantly improves its ability to represent small objects and subtle defects while suppressing background interference.

C. Global Edge Information Transfer Module Design

For the accurate detection and localisation of defect objects, edge information is crucial—the edges of defects clearly distinguish the object from its background. However, conventional object detection networks lack dedicated components to extract or utilise edge features, potentially causing the model to overlook critical contour details of the object. Under complex background noise conditions, defect edges often become blurred or conflated with the background. If the network is insensitive to edges, it may produce inaccurate detection bounding boxes or misinterpret background textures as targets. Consequently, we require a mechanism to introduce salient edge information at all scales within the feature extraction process, thereby globally enhancing the network's perception of object edges.

This paper proposes a Global Edge Information Transfer (GEIT) module, whose architecture is illustrated in Fig.4.GEIT achieves global propagation of edge information by generating multi-scale edge feature maps at shallow layers of the backbone network and fusing them with features from various scales of the backbone. Its key design components comprise multi-scale edge feature generation and edge feature fusion.

$$G_x = G_0 \otimes K_x, \quad G_y = G_0 \otimes K_y, \quad G_1 = G_x \otimes G_y \tag{5}$$

$$E^{l+1} = \mathrm{MaxPool}\left(E^l\right) \tag{6}$$

After obtaining multi-scale edge features, these are injected into the corresponding scale of the backbone to achieve

global edge sharing. We designed the ConvEdgeFusion module, which first concatenates edge features $\widetilde{E}_l$ and backbone features F_l along the channel dimension. Subsequently, cross-channel fusion is performed sequentially via 1×1 convolutions. A 3×3 convolution then emphasises local texture details and target edge shapes, followed by a final 1×1 convolution to output the enhanced feature $\hat{F}_l$.This design is repeated across layers P3–P5, achieving deep integration of edge and semantic features to enhance global contour perception and localisation accuracy.

$$\hat{F}_l = \mathrm{Conv}^{1\times1}\left(\mathrm{Conv}^{3\times3}\left(\mathrm{Conv}^{1\times1}\left(\left[F_l;\widetilde{E}_l\right]\right)\right)\right) \tag{7}$$

III. Experiments

A. Introduction to the experiment

The experimental environment was configured as follows: the GPU was an NVIDIA GeForce RTX 3090 GPU, and the operating system was Windows 11. The software environment was based on torch 1.12.0, Python 3.8. The parameters of our model are set epoch=300, the batch size is set to 16, we have an initial learning rate of 0.01 and a cosine annealing decay strategy to dynamically reduce the initial learning rate.

We employed two surface defect datasets: NEU-DET [12] and GC10-DET [13]. The NEU-DET dataset, annotated by Northeastern University, comprises 1,800 greyscale images featuring six distinct types of typical surface defects. The GC10-DET dataset contains ten different defect categories across a total of 2,280 images.

In order to comprehensively assess the performance of the model in the task of surface defect target detection, the evaluation metrics used in this study are Accuracy, Recall, Average Precision, Detection Speed, Number of Parameters and Number of Floating Point Operations (FLOPs).

B. Ablation experimental studies

To further validate the effectiveness of the proposed components, comprehensive ablation experiments were conducted on the NEU-DET and GC10-DET datasets. Starting from the baseline model, key modules were incrementally introduced, including SOAR, ISGM, GEIT, and the proposed IS-FPN, while keeping the training strategy and hyperparameter settings identical to ensure fair comparison. The detailed experimental results are reported in Tables I and Tables II.

From the results on the NEU-DET dataset, introducing the SOAR module improves the mAP_{50} from 78.1% to 80.6%, indicating that enhanced small-object-aware feature refinement contributes to better defect localization. However, this improvement comes with increased parameters and FLOPs, reflecting the additional feature processing cost. The ISGM module achieves a more balanced improvement, raising mAP_{50} to 81.3% while keeping the parameter count close to the baseline. This demonstrates that the heterogeneous attention mechanism effectively enhances cross-scale feature interaction with limited computational overhead. Similarly, the GEIT module improves mAP_{50} to 80.3% and achieves a relatively higher $mAP_{50–95}$, suggesting that explicit global edge information benefits precise localization, particularly for elongated and irregular defect boundaries.

On the GC10-DET dataset, consistent trends are observed. The SOAR module alone provides limited gains, whereas ISGM and GEIT both yield noticeable improvements in detection accuracy. This indicates that attention-guided cross-scale interaction and global edge enhancement are more robust across different defect distributions than shallow feature refinement alone.

Further experiments combining different modules show that joint optimization generally outperforms single-module configurations. For example, the combination of ISGM and GEIT yields consistent improvements on both datasets, validating their complementary roles in suppressing background noise and enhancing defect-relevant features. However, some combinations exhibit diminishing returns, highlighting the importance of carefully designed feature interaction rather than naive module stacking.

Overall Performance

The full model, integrating all proposed components, achieves the best performance on both datasets. Specifically, it reaches 83.6% mAP_{50} and 48.2% $mAP_{50–95}$ on NEU-DET, and 83.4% mAP_{50} and 47.9% $mAP_{50–95}$ on GC10-DET. Although the complete model introduces additional parameters and computational cost, the performance gains consistently outweigh the overhead, demonstrating a favorable accuracy–efficiency trade-off.

TABLE I
THE ABLATION EXPERIMENT IN THE NEU-DET DATASET.

Algorithm	mAP50	mAP(50-95)	Params/M	FLOPs/G
Baseline	78.1%	45.6%	2.6	6.6
+SOAR	80.6%	46.5%	3.1	8.6
+ISGM	81.3%	46.3%	2.7	7.1
+GEIT	80.3%	46.8%	2.8	7.7
+IS-FPN	82.1%	47.0%	3.2	7.3
+SOAR+GEIT	81.5%	46.9%	3.3	7.7
+ISGM+GEIT	81.9%	46.8%	2.9	7.6
Our	**83.6%**	**48.2%**	**3.4**	**9.2**

TABLE II
THE ABLATION EXPERIMENT IN THE NEU-DET DATASET.

Algorithm	mAP50	mAP(50-95)	Params/M	FLOPs/G
Baseline	78.1%	45.6%	2.6	6.6
+SOAR	77.7%	45.3%	3.1	8.6
+ISGM	79.9%	45.6%	2.7	7.1
+GEIT	79.3%	46.1%	2.8	7.7
+IS-FPN	81.3%	47.1%	3.2	7.3
+SOAR+GEIT	82.4%	46.9%	3.3	7.7
+ISGM+GEIT	81.0%	46.2%	2.9	7.6
Our	**83.4%**	**47.9%**	**3.4**	**9.2**

C. Comparative experimental study

In order to evaluate the effectiveness of the proposed model, we compare it with representative target detection

TABLE III
COMPARISON OF DIFFERENT TARGET DETECTION ALGORITHMS ON NEU-DET DATASET.

Algorithm	Precision%	Recall%	mAP50%	mAP(50-95)%	Params/M	FLOPs/G	FPS
Faster R-CNN [14]	33.0%	91.1%	76.9%	33.0%	138.4	368.2	36
SDD [15]	75.6%	73.8%	73.6%	39.7%	71.1	271.8	59
YOLOv3	76.3%	71.2%	76.8%	42.5%	103.7	282.2	67
YOLOv5s [16]	74.7%	74.7%	76.8%	42.4%	7.0	15.8	220
RDD-YOLO [17]	72.8%	73.0%	77.9%	42.8%	57.0	145.6	67
YOLOv6n	74.0%	71.1%	77.6%	44.5%	4.3	11.1	211
YOLOv7 [18]	76.9%	75.8%	79.2%	44.5%	36.5	103.2	82
YOLOv10n [19]	71.8%	85.4%	79.2%	37.1%	3.1	13.1	165
RT-DETR-R18 [20]	79.1%	71.8%	79.3%	46.3%	19.9	55.4	163
YOLOv8n [6]	72.2%	73.1%	77.1%	43.9%	3.0	8.2	209
YOLOv9-c [21]	79.0%	72.4%	80.7%	46.5%	25.4	103.2	175
WSS-YOLO [6]	76.0%	75.6%	82.3%	47.0%	3.2	7.7	199
YOLOv11s [22]	68.7%	70.9%	77.2%	45.6%	9.4	21.5	159
YOLOv12s [23]	72.1%	71.3%	76.2%	42.7%	9.3	21.4	107
Our	**79.2%**	**76.2%**	**82.2%**	**46.5%**	**3.4**	**9.2**	**256**

TABLE IV
COMPARISON OF DIFFERENT TARGET DETECTION ALGORITHMS ON GC10-DET DATASET.

Algorithm	Precision%	Recall%	mAP50%	mAP(50-95)%	Params/M	FLOPs/G	FPS
Faster R-CNN [14]	38.2%	59.4%	56.9%	20.4%	138.4	368.2	35
SDD [15]	66.5%	66.1%	66.2%	33.8%	71.1	271.8	51
YOLOv3	62.4%	62.7%	62.2%	32.5%	103.7	282.2	106
YOLOv5s [16]	72.4%	66.5%	69.4%	35.6%	7.0	15.8	239
RDD-YOLO [17]	72.1%	70.3%	70.4%	36.6%	57.0	145.6	98
YOLOv6n	64.3%	65.8%	66.1%	34.9%	4.3	11.1	232
YOLOv7 [18]	66.2%	71.5%	71.1%	36.1%	36.5	103.2	119
YOLOv10n [19]	76.2%	64.9%	72.1%	37.8%	6.0	13.1	208
RT-DETR-R18 [20]	72.5%	69.5%	72.6%	37.8%	19.9	55.4	137
YOLOv9-c [21]	69.0%	70.9%	71.4%	36.2%	25.4	103.2	196
WSS-YOLO [6]	66.7%	72.9%	72.0%	37.0%	3.2	7.7	185
YOLOv8n [24]	65.0%	67.2%	68.7%	36.1%	3.0	8.2	222
YOLOv11s [22]	68.3%	67.2%	68.2%	36.1%	9.4	21.5	128
YOLOv12s [23]	64.3%	68.5%	69.1%	37.3%	9.3	21.4	222
Our	**76.2%**	**78.2%**	**83.4%**	**47.2%**	**3.4**	**9.2**	**241**

algorithms, including the two-stage detector Faster R-CNN, transformer-based detectors, mainstream YOLO-series methods, and defect-oriented models such as RDD-YOLO and WSS-YOLO. All experiments are conducted under identical experimental environments and dataset settings to ensure fair comparison.

As shown in Tables III and IV, the proposed SIG-Net achieves superior detection performance on both datasets. On the NEU-DET dataset, SIG-Net reaches an mAP_{50} of 82.2% and an $mAP_{50–95}$ of 46.5%, outperforming YOLOv8n by 5.1% and 2.6%, respectively, while using fewer parameters. Compared with the advanced defect detection model WSS-YOLO, SIG-Net achieves comparable mAP_{50} with a higher inference speed.

On the GC10-DET dataset, SIG-Net further improves the mAP_{50} to 83.4% and the $mAP_{50–95}$ to 47.2%, exceeding YOLOv9-c by 12.0% and 11.0%, respectively. In addition, SIG-Net maintains a lightweight architecture with only 3.4M parameters and 9.2 GFLOPs, while achieving real-time inference at 241 FPS, demonstrating a clear advantage over heavier models such as Faster R-CNN and RT-DETR.

Overall, these results indicate that SIG-Net achieves a favorable balance between detection accuracy and computational efficiency, making it well suited for industrial defect detection tasks with strict real-time and resource constraints.

D. Generalization Experiment

To further evaluate the generalization capability of the proposed method in generic object detection tasks, comparative experiments were conducted on the PASCAL VOC2012 dataset. As shown in Table V, compared with the baseline model YOLOv11n, the proposed approach achieves stable and significant performance improvements on this dataset. Specifically, the proposed model improves mAP_{50} and $mAP_{50–95}$ by 1.5% and 1.3%, respectively, demonstrating that the method is not only effective for industrial defect detection scenarios but also exhibits strong effectiveness and generalization ability in more complex and diverse generic object detection tasks.

TABLE V
RESULTS ON GC10-DET AND VOC2012 DATASETS

Datasets	Model	mAP50	mAP(50-95)
VOC2012	YOLOv11n	62.3%	42.5%
VOC2012	**Ours**	**63.8%**	**43.8%**

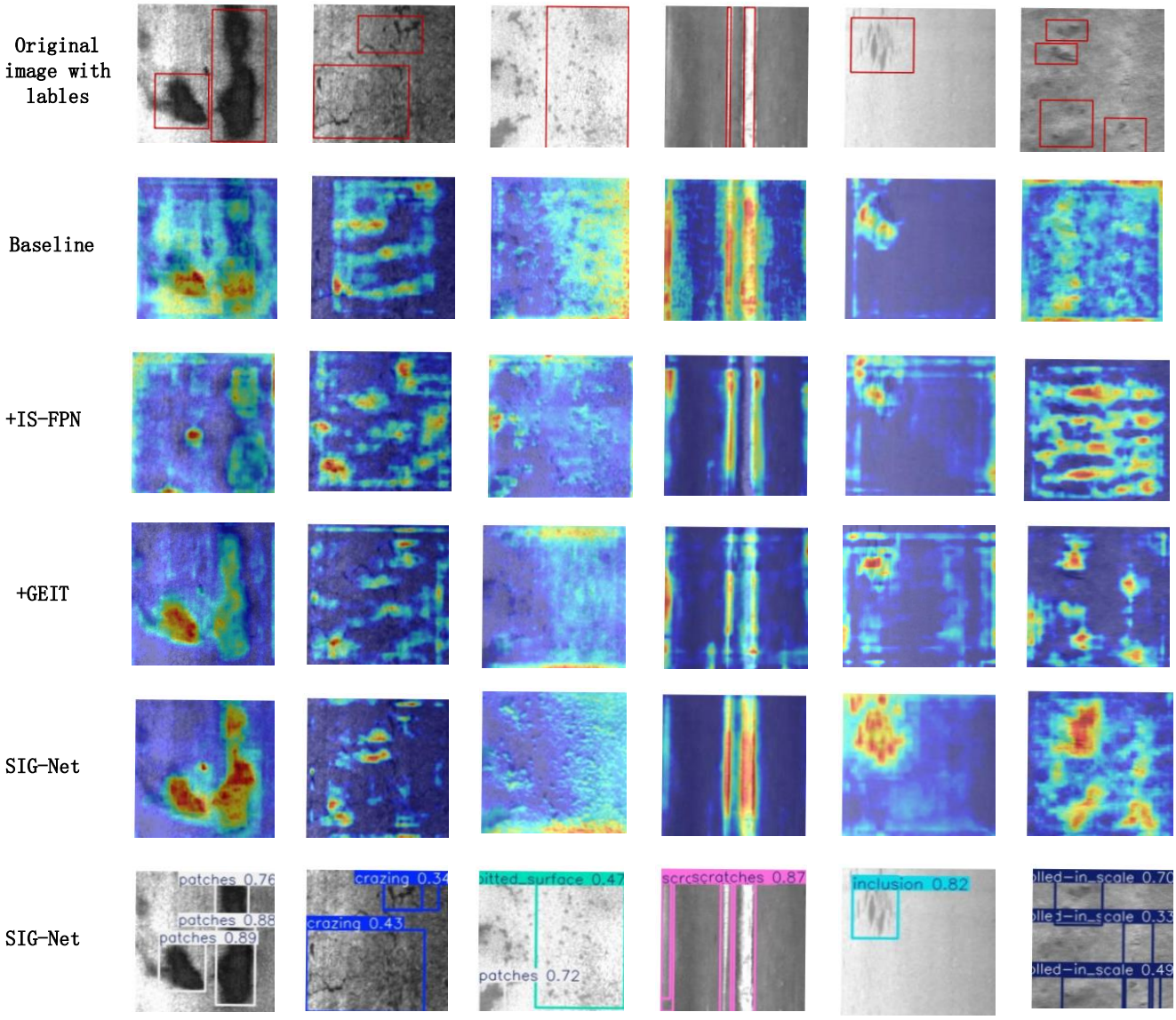

Fig. 5. Heatmaps and output maps in the NEU-DET dataset.

E. Results of the visualisation

To further validate the model's detection efficacy, Fig.5 presents ablation visualisation results, demonstrating that SIG-Net achieves more precise defect localisation while significantly reducing both missed detections and false positives.

IV. CONCLUSION

This paper proposes a surface defect detection network named SIG-Net, integrating three key modules: SOAR-FPN for enhancing small object features, ISGM for suppressing background interference through cross-scale sparse guidance, and GEIT for improving boundary perception via global edge propagation. Ablation studies validate the independent and complementary contributions of each module. Comparative experiments demonstrate that SIG-Net achieves high-precision detection under lightweight conditions on the NEU-DET dataset. The model achieves an mAP50 of 83.6% with only 3.4 million parameters, significantly outperforming existing mainstream methods. Results indicate that SIG-Net exhibits robust and efficient performance in detecting small defects within complex backgrounds. Future research will further focus on model lightweighting and generalisation capability expansion to meet broader industrial application demands.

REFERENCES

[1] G. Song, K. Song, and Y. Yan, "Edrnet: Encoder–decoder residual network for salient object detection of strip steel surface defects," *IEEE Transactions on Instrumentation and Measurement*, vol. 69, no. 12, pp. 9709–9719, 2020.

[2] J. Dong, Z. Zuo, Z. Wu, and M. Liu, "A scale-adaptive and background-robust method for surface defect detection," in *ICASSP 2025 - 2025*

IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2025, pp. 1–5.

[3] Q. Huang and J. Huang, "Comprehensive review of edge and contour detection: From traditional methods to recent advances," *Neural Computing and Applications*, vol. 37, no. 4, pp. 2175–2209, 2025.

[4] S. K. Dwivedi, M. Vishwakarma, and A. Soni, "Advances and researches on non destructive testing: A review," *Materials Today: Proceedings*, vol. 5, no. 2, pp. 3690–3698, 2018.

[5] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 779–788.

[6] M. Lu, W. Sheng, Y. Zou, Y. Chen, and Z. Chen, "Wss-yolo: An improved industrial defect detection network for steel surface defects," *Measurement*, vol. 236, p. 115060, 2024.

[7] F. Liang, Z. Wang, W. Ma, B. Liu, Q. En, D. Wang, and L. Duan, "Hdfa-net: A high-dimensional decoupled frequency attention network for steel surface defect detection," *Measurement*, vol. 242, p. 116255, 2025.

[8] M. Tan, R. Pang, and Q. V. Le, "Efficientdet: Scalable and efficient object detection," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 10 781–10 790.

[9] Y. Shi, Y. Jia, and X. Zhang, "Focusdet: an efficient object detector for small object," *Scientific Reports*, vol. 14, no. 1, p. 10697, 2024.

[10] B. Zhao, Y. Chen, X. Jia, and T. Ma, "Steel surface defect detection algorithm in complex background scenarios," *Measurement*, vol. 237, p. 115189, 2024.

[11] R. Sunkara and T. Luo, "No more strided convolutions or pooling: A new cnn building block for low-resolution images and small objects," in *Joint European conference on machine learning and knowledge discovery in databases*. Springer, 2022, pp. 443–459.

[12] Y. He, K. Song, Q. Meng, and Y. Yan, "An end-to-end steel surface defect detection approach via fusing multiple hierarchical features," *IEEE transactions on instrumentation and measurement*, vol. 69, no. 4, pp. 1493–1504, 2019.

[13] X. Lv, F. Duan, J.-j. Jiang, X. Fu, and L. Gan, "Deep metallic surface defect detection: The new benchmark and detection network," *Sensors*, vol. 20, no. 6, p. 1562, 2020.

[14] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," *IEEE transactions on pattern analysis and machine intelligence*, vol. 39, no. 6, pp. 1137–1149, 2016.

[15] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "Ssd: Single shot multibox detector," in *European conference on computer vision*. Springer, 2016, pp. 21–37.

[16] G. Jocher, A. Chaurasia, A. Stoken, J. Borovec, Y. Kwon, K. Michael, J. Fang, C. Wong, Z. Yifu, D. Montes *et al.*, "ultralytics/yolov5: v6. 2-yolov5 classification models, apple m1, reproducibility, clearml and deci. ai integrations," *Zenodo*, 2022.

[17] C. Zhao, X. Shu, X. Yan, X. Zuo, and F. Zhu, "Rdd-yolo: A modified yolo for detection of steel surface defects," *Measurement*, vol. 214, p. 112776, 2023.

[18] C.-Y. Wang, A. Bochkovskiy, and H.-Y. M. Liao, "Yolov7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2023, pp. 7464–7475.

[19] A. Wang, H. Chen, L. Liu, K. Chen, Z. Lin, J. Han *et al.*, "Yolov10: Real-time end-to-end object detection," *Advances in Neural Information Processing Systems*, vol. 37, pp. 107 984–108 011, 2024.

[20] Y. Zhao, W. Lv, S. Xu, J. Wei, G. Wang, Q. Dang, Y. Liu, and J. Chen, "Detrs beat yolos on real-time object detection," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2024, pp. 16 965–16 974.

[21] C.-Y. Wang, I.-H. Yeh, and H.-Y. Mark Liao, "Yolov9: Learning what you want to learn using programmable gradient information," in *European conference on computer vision*. Springer, 2024, pp. 1–21.

[22] R. Khanam and M. Hussain, "Yolov11: An overview of the key architectural enhancements," *arXiv preprint arXiv:2410.17725*, 2024.

[23] Y. Tian, Q. Ye, and D. Doermann, "Yolov12: Attention-centric real-time object detectors," *arXiv preprint arXiv:2502.12524*, 2025.

[24] J. Glenn, "YOLOv8 release v8.1.0," https://github.com/ultralytics/ultralytics/releases/tag/v8.1.0, Mar. 2024, accessed: 2024-03-07.