Transfer-Learned Lightweight YOLOv8 for On-Site Defect Detection of Tower-Crane Traction Wire Ropes with BIM-Linked Safety Management

Yongchao Xie[1,2], Wenhao Li[2], Chenrun Dong[2], Yichuan Deng[2*]

[1] Guangzhou Hengsheng Construction Group Co., Ltd
[2] School of Civil Engineering and Transportation, South China University of Technology,Guangzhou,China
ctycdeng@scut.edu.cn

Abstract –

With the increasing number of high-rise construction projects, the health status of tower-crane traction wire ropes is directly linked to the safety of hoisting operations. Addressing the limitations of traditional detection methods—such as manual visual inspection, magnetic induction, and ultrasonic testing, which suffer from low frequency, high subjectivity, and limited coverage—this paper proposes an on-site vision-based inspection method for tower-crane traction wire ropes, which can be integrated as a service into a BIM-based safety management platform. The proposed module integrates the YOLOv8 object detection framework with lightweight feature enhancement modules, including GSConv, SEBlock, and SimAM, to construct eight combined model configurations. Using a transfer learning strategy, the models pre-trained on a public cable dataset were adapted to a tower-crane wire-rope dataset. Experimental results demonstrate that the optimal combination (GSConv+SEBlock+SimAM) exhibits excellent generalization under small-sample conditions, achieving a 1.58% improvement in mAP@0.5:0.95, and optimizing training box loss, classification loss, and distribution focal loss by 0.0477, 0.0015, and 0.0900, respectively. Furthermore, transfer learning significantly enhanced model robustness in complex backgrounds; compared to the non-transferred model, the transferred model increased Precision by 0.5092 and mAP@0.5 by 15.07%, effectively resolving the difficulty in identifying strand slackening defects. In engineering deployment, the inferred defect events can be linked to rope-related BIM components for visualization and maintenance traceability.

Keywords –

Defect Detection; Deep Learning; YOLO; Tower Crane;BIM

1 Introduction

Tower cranes are essential for high-rise construction due to their large working radius and lifting efficiency. Their operational safety strongly depends on the condition of traction wire ropes that carry loads during lifting and positioning. In harsh site environments (wind, rain, dust abrasion, corrosion, and frequent start–stop impacts), ropes are prone to wear, corrosion, broken wires, and strand slackening. Undetected early defects may accumulate and cause severe failures such as rope rupture and falling objects. Accordingly, international and national standards specify rope care, inspection, and discard criteria, highlighting the need for reliable defect identification and timely safety decisions [1].

Despite these requirements, tower-crane rope inspection still largely relies on periodic manual checks and experience-based judgment, which are limited by poor accessibility and inconsistent quality. Conventional nondestructive testing (NDT) methods (e.g., MFL, eddy current, ultrasonic) can quantify defects in controlled settings but often face deployment burden, discontinuous coverage, and sensitivity to lift-off and environmental interference on construction sites. For instance, MFL-based inspection can detect broken wires quantitatively, yet robustness under complex operational conditions remains challenging [2].

With advances in computer vision, image-based non-contact inspection has become an attractive option for industrial defect detection, reducing dependence on manual expertise. Reviews on deep-learning-based surface defect detection indicate that data-driven models can improve robustness over rule-based pipelines, especially under cluttered backgrounds and illumination changes. However, direct application to tower-crane traction ropes is still difficult because defect cues are typically small and subtle, site conditions vary significantly, and labeled tower-crane rope data are limited—factors that can hurt generalization and increase both false alarms and missed detections [3].

In this context, one-stage detectors in the YOLO family are widely used for real-time inspection due to a favorable speed–accuracy trade-off [4]. Building on a YOLOv8 baseline, this paper develops an enhanced method for detecting typical traction-rope surface defects (broken wires and strand slackening) under complex construction conditions. We strengthen feature representation using lightweight components—GSConv for efficient convolution, SE blocks for channel recalibration, and SimAM for parameter-free attention—to emphasize weak defect cues [5][6][7]. To further address limited tower-crane samples and improve training stability and generalization, we employ transfer learning by adapting knowledge from a related cable/wire-rope dataset to tower-crane traction-rope imagery [8]. Overall, the proposed framework aims to bridge the gap between standard-driven inspection needs and site-deployable, accurate, and efficient defect detection for tower-crane traction wire ropes [9]. Figure 1 shows the overall framework of the proposed method.

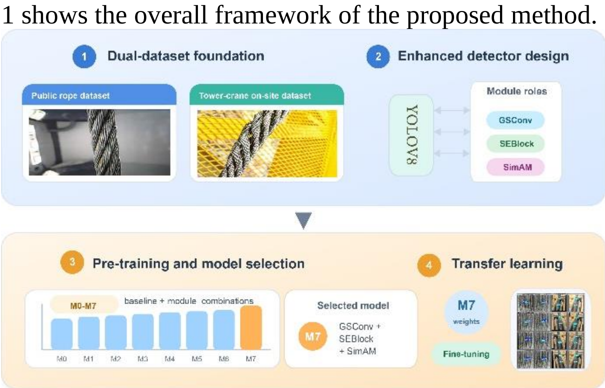


Figure 1 Overall framework of the proposed method

2 Related Work

Surface defect detection methods based on computer vision can be broadly grouped into traditional image processing methods and deep learning-based methods. Traditional methods rely on hand-crafted rules or templates, extracting low-level cues (e.g., grayscale, edges, textures) and comparing them with thresholds or references to detect anomalies. They were widely used in early industrial automation due to low computational cost, but are highly sensitive to lighting, background clutter, and viewpoint changes. Moreover, they often fail on defects with weak textures or irregular patterns, limiting robustness under variations in defect shape, size, and location.

In recent years, deep learning-based methods have become a strong alternative. Instead of manual feature design, multi-layer neural networks learn discriminative representations from data for classification and detection. Such models show higher adaptability to diverse defect types, complex backgrounds, and changing imaging conditions, and have achieved strong accuracy and robustness in practical industrial inspection. With improved imaging devices and growing industrial data, deep learning has become the dominant direction for surface defect detection.

2.1 Deep Learning

Deep learning, a key branch of machine learning, is now central to computer vision and related fields. First introduced by LeCun et al. [1], it uses hierarchical neural networks to automatically learn features from perception to decision. In 2012, Krizhevsky et al. [11][12] proposed AlexNet, which substantially reduced ImageNet classification error and accelerated practical adoption. Subsequent advances, such as the ReLU activation function [13], further improved convergence and alleviated gradient vanishing.

Convolutional neural networks (CNNs) remain the workhorse for image recognition, segmentation, and object detection [14]. Although newer paradigms such as vision transformers based on self-attention [15] and diffusion models [15] continue to emerge, CNNs are still widely adopted in industrial defect inspection due to their strong local feature modeling and reliable generalization.

2.2 Surface Defect Detection Based on Deep Learning

Given the limitations of traditional pipelines, deep learning has been increasingly applied to surface defect detection. By learning from image datasets, these methods can model complex, nonlinear defect patterns and maintain performance under noise, weak features, and variable defect shapes, making them suitable for more diverse industrial scenarios.

However, several gaps remain for reliable deployment in tower-crane traction wire ropes under real construction conditions. First, many high-performing systems are developed under relatively controlled imaging assumptions, whereas construction sites exhibit strong domain shifts due to clutter, reflections, shadows, weather, and frequent viewpoint changes, which can destabilize false-alarm and miss rates when the deployment domain differs from training data [16]. Second, early-stage defects such as initial broken wires and strand slackening are often small, low-contrast, and weakly textured, and can be suppressed by downsampling or multi-scale fusion; improving sensitivity to these subtle defects may conflict with real-time requirements. Third, tower-crane defect data are typically limited and imbalanced, and high-quality bounding-box annotation is expensive. While anomaly detection has advanced rapidly (e.g., MVTec AD [17], PaDiM [18], PatchCore [19]),these methods often focus on one-class “normal vs. abnormal” detection and may not directly provide the defect-type categorization needed for maintenance decisions.

To address these issues, this study adopts an object-detection formulation aligned with practical inspection requirements and builds on a real-time YOLOv8-style framework, while enhancing weak defect representation with lightweight modules. Specifically, GSConv improves efficient feature extraction, SE blocks recalibrate channel importance, and SimAM provides parameter-free attention to emphasize small and subtle defect cues without excessive computational overhead. Furthermore, transfer learning is used to mitigate small-sample instability and domain-shift effects by leveraging feature priors from related cable/wire-rope imagery and fine-tuning on tower-crane traction-rope data.

3 Method

In this work, the YOLOv8 architecture was enhanced by integrating three lightweight feature enhancement modules: GSConv, SEBlock, and SimAM. These modules were incorporated into the network to improve the model's ability to detect defects in wire ropes, particularly under conditions with limited sample sizes. The model was first pre-trained on a publicly available cable wire rope dataset and then fine-tuned on a custom tower crane wire rope dataset using transfer learning techniques. This transfer learning strategy was adopted to address the challenges of small sample data and to optimize model performance for tower crane-specific wire rope defects.

3.1 Feature Enhancement Modules

- **GSConv [7]**(Ghost-Shuffle Convolution): GSConv is a lightweight convolutional module designed to enhance feature extraction efficiency while reducing computational complexity. It combines the Ghost module’s feature redundancy generation mechanism with ShuffleNet's channel shuffling strategy, enabling efficient interaction between channels. This improves feature representation while maintaining low computational cost.
- **SEBlock [5]**(Squeeze-and-Excitation Block): SEBlock is a channel attention mechanism that dynamically adjusts the importance of each channel in the feature map. It operates in two stages: the squeeze stage, where global average pooling is applied to reduce the spatial dimensions and generate a global channel descriptor, and the excitation stage, where a fully connected network generates channel-wise weights. These weights are used to scale the input feature map, thereby enhancing the model's focus on the most relevant features.
- **SimAM [6]**(Simple Attention Module): SimAM is a lightweight spatial attention mechanism designed to enhance the model's ability to focus on significant regions in the feature map. Without introducing additional parameters, SimAM computes an energy score for each pixel based on its deviation from the overall distribution of the feature map. This score is then used to generate attention weights, which are applied to the feature map to improve spatial localization of important features.

3.2 Insertion of Each Module

Each of the three enhancement modules was inserted into the YOLOv8 network at strategic locations to maximize their impact on defect detection performance:

1. GSConv was inserted into the first convolutional layer of the Backbone (Layer 1) after the initial downsampling. This location allows the module to enhance the feature extraction process at an early stage, improving the efficiency of convolutional operations and reducing the computational burden. The module insertion method is shown in Table 1。
2. SEBlock was inserted after the first C2f module in the Backbone (Layer 2) and after the second C2f module in the Head (P3'). This insertion improves the channel-level feature attention at crucial points in the network, enhancing the model's focus on important defect features, especially those with small or weak targets.
3. SimAM was inserted in tandem with SEBlock, following the same strategy in both the Backbone (Layer 2) and the Head (P3'). This allows the spatial attention mechanism to work alongside the channel attention mechanism, improving the focus on spatially significant regions and enhancing the model's ability to detect defects localized in specific areas.

3.3 Model Design Principles

The model was designed with several key principles to ensure efficient integration of the enhancement modules while maintaining the integrity of the original YOLOv8 architecture:

- **Position Stability**: Each module was inserted at positions in the network where it would have the greatest impact on feature enhancement without disrupting the flow of information. This approach ensures that the enhancements do not interfere with the network’s core operations.
- **Minimal Invasion**: The insertion of each module was done with minimal disruption to the existing

network architecture. Only the most relevant layers were modified or enhanced, ensuring the network's overall structure and performance remained intact.

- **Modularity and Ablation Independence**: The modules were designed to function independently of each other, allowing for flexible experimentation and ablation studies to assess their individual contributions to the model's performance.
- **Semantic Consistency**: The modules were deployed at layers with similar semantic levels to ensure that feature representations remain consistent across the network. This helps to maintain the integrity of the network's feature extraction and ensures that the enhancements are applied effectively.

4 Experiment and Results

4.1 Data Collection and Annotation

In this study, we constructed two datasets to support pre-training and domain-specific fine-tuning, respectively: (1) a publicly available cable wire-rope defect dataset from Roboflow Universe and (2) a self-collected tower-crane traction wire-rope dataset captured in real construction environments. The public dataset was used to provide the detector with generalizable rope texture and defect representations prior to domain adaptation; it contains **4,317 images** with existing annotations. The dataset is shown in Figure 2. Self-built tower-crane traction wire-rope dataset containing **260 images** was used as the target-domain dataset for small-sample transfer learning.

For the dataset, the data pipeline was designed to reflect practical on-site conditions and to control annotation quality. First, raw images were acquired on-site under diverse illumination (backlight, shadow, strong reflection), weather, and background clutter, with multiple viewing angles and shooting distances to increase scene diversity. Because the original recordings contain redundant frames and substantial quality variation, we performed a structured preprocessing workflow before labeling: (i) frame extraction and de-duplication to remove near-identical samples; (ii) quality screening to exclude heavily blurred or severely occluded frames; (iii) resolution normalization to a consistent input size (the long side was resized to 640 px) to match the detector's training configuration; and (iv) format standardization to ensure each image has a one-to-one label file for downstream YOLO training..

Annotation was conducted fully manually using LabelImg, with rectangular bounding boxes drawn tightly around defect regions. To reduce label noise (a frequent cause of performance instability in small datasets), we adopted a multi-step verification procedure: each labeled image was checked for (i) correct class assignment, (ii) tightness of the bounding box boundary to minimize excessive background inclusion, and (iii) completeness when multiple defects appear in the same image. All annotations were exported in YOLO text format, and label files were automatically validated to prevent missing labels, duplicated boxes, or invalid coordinates. This annotation-and-QC process is labor-intensive but essential for training a detector to localize subtle, small-scale defects reliably in complex construction scenes.

To explicitly account for the limited amount of defect data and the strong domain shift in construction environments, we applied an offline augmentation pipeline after manual labeling. For each selected original image, we generated multiple augmented variants using mild geometric/photometric perturbations together with stronger corruption-oriented transformations to emulate harsh on-site imaging conditions. Importantly, the corresponding bounding boxes were transformed consistently with each augmentation operation, producing a fully aligned augmented training set.

4.2 Data Augmentation and Splitting

To prevent overfitting and increase the robustness of the model, various data augmentation techniques were employed during the training process. Data augmentation was implemented in two stages:

- **Light Augmentation**: In this initial stage, simple transformations were applied to the images to simulate common variations encountered in real-world scenarios. These transformations included random horizontal flipping, rotation (±20°), and slight color jittering (adjusting brightness, contrast, and saturation). Additionally, random cropping and resizing were applied to simulate changes in viewpoint and focus.
- **Aggressive Augmentation**： The second stage of augmentation aimed to simulate more extreme variations. Techniques like Gaussian noise injection were applied to simulate low-light conditions and image sensor noise. Strong lighting conditions and lens distortion were also simulated using lens effect augmentation, making the model more resilient to such variations. These augmentations ensured that the model would not overfit to specific image patterns and could generalize well across different conditions.

4.3 Experimental Setup and Results

In this section, we present the results of two experiments conducted to evaluate the proposed model for defect detection in tower crane wire ropes. The first experiment focuses on evaluating the various combination models' performance on the publicly available cable wire rope dataset, while the second experiment tests the model on the custom tower crane wire rope dataset collected in real-world construction environments. Both experiments are analyzed in terms of performance metrics such as precision, recall, and mean average precision (mAP), and the results are compared to baseline models.

4.3.1 Comparison of Combination Model Training

To evaluate the impact of the proposed feature enhancement modules (GSConv, SEBlock, and SimAM) on the performance of the YOLOv8 model, we conducted a model comparison experiment by setting up seven different combinations of these modules. The goal of this experiment was to identify the contribution of each individual module as well as the combined effect of all three modules on the model's detection performance.

Each model was trained using the same tower crane wire rope dataset, and the evaluation metrics (precision, recall, and mAP) were computed for each configuration.

The experimental results for each model configuration are summarized in **Table 1**. The results show that the introduction of the feature enhancement modules consistently improved the model's performance compared to the baseline YOLOv8 model.

The comparison experiment clearly shows that each of the three enhancement modules—GSConv, SEBlock, and SimAM—contributes to improved detection performance. Among them, the combination of all three modules led to the highest performance, achieving the best precision, recall, and mAP metrics. The combination of GSConv and SEBlock resulted in the highest performance among two-module configurations, but the full model—incorporating GSConv, SEBlock, and SimAM—achieved the best overall detection results. This suggests that the feature extraction, channel attention, and spatial attention mechanisms work synergistically to improve the model's ability to detect defects.

Therefore, this study ultimately selects M7 as the target model for subsequent transfer learning and inference deployment, and based on this, conducts structural transfer and domain adaptation experiments for tower crane wire rope target detection tasks.

4.3.2 Application of Transfer Learning in Tower Crane Wire Rope Detection

In this section, transfer learning was applied to enhance the model's performance on the tower crane wire rope dataset, which is relatively small compared to publicly available datasets. The transfer learning strategy adopted here involved loading the weights of the M7 model.

To evaluate the impact of the transfer learning strategy, we compared the performance of the model initialized with M7 weights (transfer learning) to that of a model trained from scratch on the tower crane wire rope dataset. The comparison aimed to demonstrate the advantages of leveraging pre-trained knowledge over starting with random weights. Table 2 provides a direct comparison between the transfer learning model and the scratch-trained model. The transfer learning model outperformed the scratch-trained model in all evaluation metrics, showing a significant advantage in terms of detection performance.

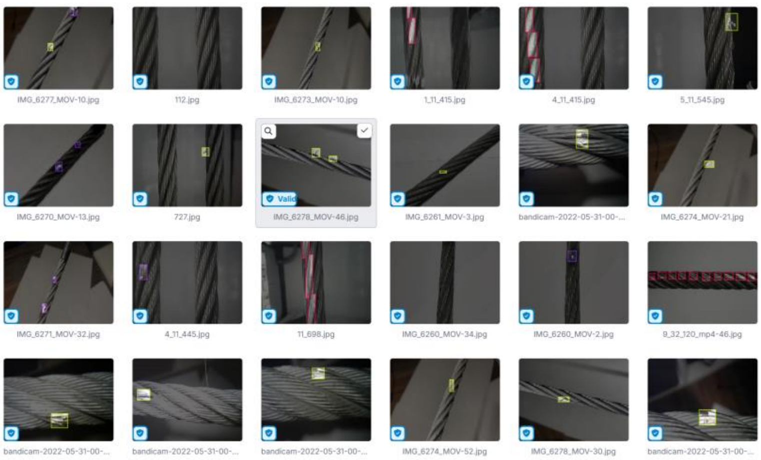


Figure 2 A Publicly Available Cable Wire-rope Defect Dataset

Table 1 Comparison of performance indicators among various models

Model	Precision	Recall	F1-score	mAP@0.5	mAP@ 0.5:0.95
M0(Base)	0.8521	0.7720	0.8101	0.8165	0.3710
M1(GSConv)	0.8594	0.7638	0.8088	0.8163	0.3853
M2(SEBlock)	0.8470	0.7682	0.8057	0.8163	0.3750
M3(SimAM)	0.8550	0.7652	0.8076	0.8149	0.3694
M4(GS+SE)	0.8464	0.7450	0.7925	0.8091	0.3744
M5(GS+Sim)	0.8501	0.7618	0.8038	0.8134	0.3719
M6(SE+Sim)	0.8491	0.7641	0.8041	0.8137	0.3725
M7(GS+SE+Sim)	0.8581	0.7689	0.8108	0.8187	0.3868

Table 2 Comparison of Transfer Learning Model vs. Scratch Training Model

Model	Precision	Recall	mAp@0.5	Loss
Transfer Learning Model	0.969	0.225	0.310	8.175
Scratch Training Model	0.366	0.213	0.235	8.240

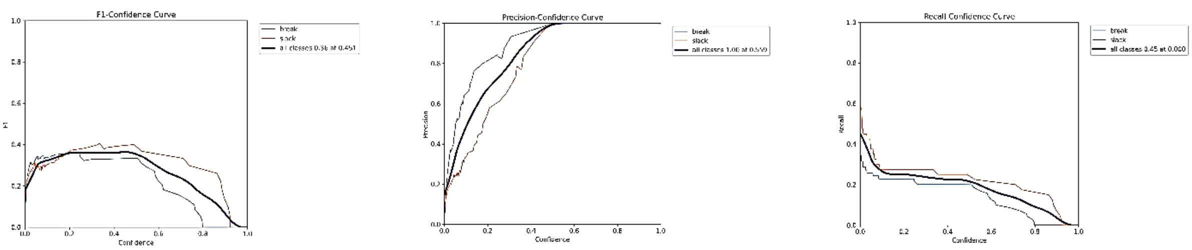
To analyze the reported high-precision/low-recall phenomenon, we further evaluated the transferred model under different confidence thresholds. As shown in Figure 3, increasing the threshold consistently improves precision but reduces recall. The best F1 score is achieved at a threshold of 0.451, indicating that threshold tuning can partially improve the precision-recall balance. Nevertheless, the maximum recall remains limited even at very low thresholds, suggesting that the recall deficiency is not caused solely by threshold setting, but is also related to the small-sample target-domain condition and the domain gap.

To visually assess the effectiveness of transfer learning in wire rope defect detection, a qualitative analysis was conducted on a batch of 16 images from the validation set. This analysis compared the prediction results of the model with and without pre-training using the same input images, and contrasted these predictions with the corresponding label images.

Figure 4 shows the label images, which provide the ground truth annotations for each defect in the wire ropes. Figure 5 presents the prediction results of the model trained from scratch, while Figure 8 illustrates the prediction results of the model initialized with pre-trained weights from the M7 model.

In the qualitative comparison, the model trained from scratch (Figure 5) displayed lower accuracy in detecting subtle defects such as strand slackening and wire breakage, especially in cases where the defects were partially occluded or in challenging backgrounds. On the other hand, the model using transfer learning (Figure 6) demonstrated more precise defect localization and better handling of occlusions and background noise, showing significant improvements in both detection accuracy and localization.

This qualitative analysis confirms that transfer learning contributes to more reliable and accurate defect detection, especially in real-world, complex environments where defects are often subtle and challenging to detect. The model initialized with pre-trained weights achieved better alignment with the ground truth annotations, validating the effectiveness of the transfer learning strategy for wire rope defect detection



(a) F1-confidence curve (b) Precision-confidence curve (c) Recall-confidence curve

Figure 3. Confidence-threshold sensitivity analysis of the transferred model on the tower-crane target dataset

4.3.3 BIM-Oriented Application within a Safety Management Platform.

To support project-level BIM deliverables, the proposed defect detector is packaged as an inspection service that can be connected to a BIM-based safety management platform. In deployment, each inference result is structured as a "defect event" containing the defect type (broken wire/strand slackening), bounding box, confidence score, timestamp, and the associated image evidence. The platform maintains a rope–crane component registry, where defect events are associated with rope-related BIM components via crane ID and component identifiers, enabling defect traceability at the component level. The linked events can be visualized in the BIM viewer and exported as maintenance records to support periodic inspection reporting. In addition, the event log provides a basis for rule-based warning policies, allowing a practical workflow of "automatic alert + human verification" that aligns with on-site safety management needs.

Figure 4.A batch label image

Figure 5. A batch without pre trained prediction results

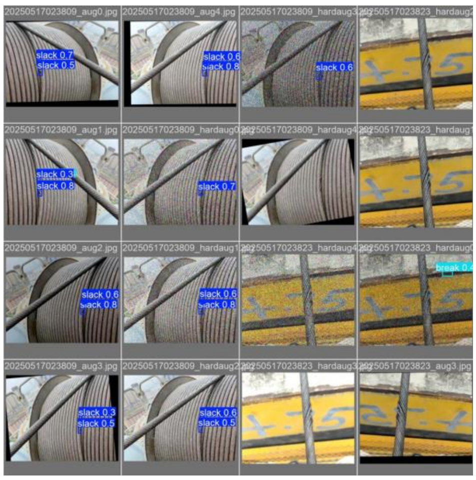


Figure 6. A batch with pre trained prediction results

5 Discussion

In the tower-crane traction rope experiments, the transfer-learning model exhibits a clear "high precision–low recall" profile (Precision = 0.969, Recall = 0.225), indicating a conservative decision boundary that effectively suppresses false alarms caused by stains, specular reflections, or background structures, but still misses a portion of subtle defects—especially wire-breakage patterns that appear as weak, small-scale texture discontinuities under long-distance imaging and vibration-induced blur. This behavior reduces unnecessary stoppages in practice, yet it also suggests that safety-critical deployment should adopt an "automatic warning + human verification" workflow and prioritize recall improvements via small-object-sensitive design.

Meanwhile, the GSConv–SEBlock–SimAM combination (M7) demonstrates a synergistic effect: efficient feature extraction with preserved rope texture (GSConv), channel-wise emphasis on defect-related responses (SEBlock), and parameter-free spatial focusing (SimAM), which together improves convergence and stability in cluttered scenes.

Finally, given the small-sample constraint of the self-built dataset, transfer learning provides useful priors for linear-structure and metal-fatigue textures, improving robustness under strong augmentations; qualitatively, slackening is generally easier to detect than micro-scale broken wires because it changes the rope's macroscopic geometry more noticeably.

6 Conclusion

This study proposed an intelligent defect detection method for tower-crane traction wire ropes based on the YOLOv8 framework, integrating three lightweight enhancement modules (GSConv, SEBlock, and SimAM). Extensive quantitative and qualitative experiments on both the cable wire-rope dataset and the self-built tower-crane wire-rope dataset demonstrated that these modules significantly improve detection performance, particularly under complex backgrounds and for subtle defects, and that the full model achieves the best overall results, confirming the synergistic benefits of the combined design. Moreover, transfer learning further strengthens robustness compared with training from scratch, enabling more reliable localization and recognition in challenging scenes and supporting the feasibility of real-time on-site inspection.

In practice, the proposed detector can be deployed as an inspection service and further connected to BIM-based safety management workflows to support traceable maintenance decisions.

Acknowledgments

The authors would like to acknowledge the support from the National Natural Science Foundation of China (52308314), the Guangdong Basic and Applied Basic Research Foundation (2023A1515030169), and the Guangzhou Science and Technology Project (2025A04J5216). This work was also supported by the Science and Technology Program of Guangzhou Municipal Construction Group Co., Ltd. ([2023]-KJ023, [2024]-KJ043, [2024]-KJ047).

References

[1] ISO. *ISO 4309:2017 Cranes — Wire ropes — Care and maintenance, inspection and discard*. International Organization for Standardization, 2017.

[2] Zhang, J., et al. “Quantitative Inspection of Remanence of Broken Wire Rope Based on Compressed Sensing.” *Sensors*, 16(9), 1366, 2016.

[3] Ameri, R., et al. “A systematic review of deep learning approaches for surface defect detection in industrial applications.” *Expert Systems with Applications*, 2024.

[4] Redmon, J., Divvala, S., Girshick, R., Farhadi, A. “You Only Look Once: Unified, Real-Time Object Detection.” In *CVPR*, 2016.

[5] Hu, J., Shen, L., Sun, G. “Squeeze-and-Excitation Networks.” In *CVPR*, 2018.

[6] Yang, L., Zhang, R.-Y., Li, L., Xie, X. “SimAM: A Simple, Parameter-Free Attention Module for Convolutional Neural Networks.” In *ICML (PMLR 139)*, 2021.

[7] Li, H., et al. “Slim-neck by GSConv: A lightweight-design for real-time detector architectures.” *arXiv:2206.02424*, 2022.

[8] Pan, S. J., Yang, Q. “A Survey on Transfer Learning.” *IEEE Transactions on Knowledge and Data Engineering*, 22(10), 1345–1359, 2010.

[9] Ultralytics. “YOLOv8 Model Documentation (released 2023).”

[10] LeCun Y, Bengio Y, Hinton G. Deep learning. Nature. 2015 May 28;521(7553):436-44. doi: 10.1038/nature14539.

[11] Krizhevsky, A., Sutskever, I., Hinton, G. E. *ImageNet Classification with Deep Convolutional Neural Networks.* NeurIPS, 2012.

[12] Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., Fei-Fei, L. *ImageNet: A Large-Scale Hierarchical Image Database.* CVPR, 2009.

[13] Glorot X, Bordes A, Bengio Y. Deep sparse rectifier neural networks.Proceedings of the fourteenth international conference on artificial intelligence and statistics. JMLR Workshop and Conference Proceedings, 2011: 315-323.

[14] Jian Li. Research progress on surface defect detection technology based on deep learning. "Computer Integrated Manufacturing Systems 30.3 (2024)

[15] Dosovitskiy, A., Beyer, L., Kolesnikov, A., et al. *An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale.* ICLR, 2021.F. -A. Croitoru, V. Hondru, R. T. Ionescu and M. Shah, "Diffusion Models in Vision: A Survey," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 9, pp. 10850-10869, 1 Sept. 2023, doi: 10.1109/TPAMI.2023.3261988.

[16] Li, J., Li, H., Hu, X., et al. Research progress of surface defect detection technology based on deep learning. *Computer Integrated Manufacturing Systems*, 30(3), 2024.

[17] Bergmann, P., Fauser, M., Sattlegger, D., Steger, C. MVTec AD—A Comprehensive Real-World Dataset for Unsupervised Anomaly Detection. *CVPR*, 2019.

[18] Defard, T., Setkov, A., Loesch, A., Audigier, R. PaDiM: A Patch Distribution Modeling Framework for Anomaly Detection and Localization. *ICPR Workshops / Springer* (and arXiv:2011.08785), 2021.

[19] Roth, K., Pemula, L., Zepeda, J., Schölkopf, B., Brox, T., Gehler, P. Towards Total Recall in Industrial Anomaly Detection. *CVPR*, 2022. (PatchCore)