Journal of Applied Automation Technologies

Research Paper

Real-Time Defect Detection in Intelligent Manufacturing Systems Based on YOLOv8

Alessio Moretti[1] and Luca Ferrari[1, *]

[1] Department of Computer Science, University of Milan, Physics and Computer Science, 20-133 Milan, Italy
*Corresponding author: luca.fe@studenti.unimi.it

Abstract. In high-end manufacturing, deep learning-based detection systems are gradually being used for automated quality control. In order to create a fast, real-time defect detection system, a robust backbone network and a powerful data preparation module were established in this study. The large intra-class variability, dynamic imaging environment, and strict requirements for real-time inference are serious issues faced by current work in industrial defect detection. Geometric transformations, photometric variations, and synthetic defect generation are types of data augmentation techniques that can enhance the model's generalization ability. To ensure stable convergence and avoid overfitting, the model training will use an adaptive learning rate scheduler, a composite loss function, and reasonable regularization methods. Architecture optimization: Multi-scale feature fusion and anchor-free detection heads improve the sensitivity to sparse surface defect recognition. Using the same method across multiple industries, each with its own products and defects. According to the experimental results, the average accuracy, recall rate, and processing speed of these two frameworks are all above average. The new pipeline runs stably on GPU servers and in edge cases. By using error and ablation analysis, the individual impact of the new components was revealed, providing a reference for future research. In summary, the aforementioned research indicates that the integrated framework can meet the high demands of large-scale automated quality inspection in modern manufacturing.

Keywords: *Industrial Automation, Deep Learning, Defect Detection, YOLOv8, Data Augmentation, Quality Control*

Received on 29 June 2025, Accepted on 26 December 2025, Published on 26 March 2026

Introduction

The application of technologies such as intelligent technology, the Internet of Things (IoT), and big data analytics has recently transformed global production [1]. Intelligent manufacturing systems based on Industry 4.0 achieve high efficiency, flexibility, and reliability through cyber-physical mechanisms and extensive automation [2]. With the increasing demand for customized products and high quality, manufacturers are paying more and more attention to defects. Therefore, we are striving to improve product quality and competitiveness while reducing costs [3]. Accurate testing can prevent products from wasting resources due to quality issues and being released to the market [4]. Due to the large number of personnel, human errors, and low scalability, manual inspection is still widely used in certain areas [5]. Rule-based algorithms and traditional machine vision can be partially automated, but they cannot adapt to complex real-world environments and various dynamic or defect backgrounds [6]. With the development of modern factories, many new products have emerged, and these products need to meet various lighting requirements and have high-speed operation capabilities. The demand for automated, reliable, real-time inspection equipment is increasing [7]. Many companies have already begun using intelligent vision technology to improve throughput, accuracy, and robustness [8].

The development of convolutional neural networks and deep learning has led to the introduction of many new defect detection methods [9]. New deep models achieve strong robustness to complexity and noise, as well as powerful feature extraction capabilities [10]. Among them, the YOLO (You Only Look Once) series is highly praised in the industry for its fast detection and single-step detection [11]. The early versions of YOLO had some shortcomings; they were unable to detect small, frequent, or rare defects, and could be unreliable in complex visual environments in factories [12]. With the development of object detection technology, multi-scale feature processing and various network components have recently been improved [13]. YOLOv8 is the latest version, with improvements in structure and training enhancing detection accuracy, speed, and generalization ability [14]. The aforementioned research indicates that to improve the performance of different industrial surfaces, data augmentation and adaptive training are necessary [15].

After some improvements, high defect variability, visual domain transfer, and strict real-time requirements are still present in production. This paper introduces an all-weather real-time defect detection system based on YOLOv8 for smart manufacturing. Improved data augmentation and training methods to enhance the model's robustness; designed an optimized structure to handle small defects more flexibly while maintaining high inference speed; and conducted comprehensive testing in accordance with industry standards to demonstrate the improved accuracy and generality. The following section will briefly introduce the current detection methods and their improvements to YOLOv8, detail the enhancements made to this project, present comprehensive experiments, and discuss its applications in industrial settings.

Review of Defect Detection Algorithms

Conventional Methods

Fixed-rule machine vision systems or manual visual inspections are the primary methods for detecting defects in production. Historically, human inspectors have checked for defects such as scratches, cracks, or improper assembly in components. Sometimes, light sources and magnifying glasses are used to improve detection accuracy. Although workers can adapt to minor differences in production line and environmental changes, this method is limited by subjectivity, fatigue, and inconsistency as the production line becomes more complex and high-speed [16]. The scope of manual inspection is limited, and due to inconsistent attention or cognitive overload, systematic errors can also occur.

To address the aforementioned issues, traditional machine vision has begun using rule-based algorithms (such as template matching, threshold processing, and edge detection) to replace some manual work. These systems inspect changes in geometric features and intensity to detect potential defects. They simultaneously improve speed and repeatability [17]. Traditional methods have some drawbacks, so feature selection and environmental control need to be precise. Change the lighting, angle, or other positions of the object to reduce its recognition rate. The aforementioned methods may be too unflexible if there are various backgrounds, types of defects, or very small defects in the manufacturing environment [18]. To meet the demands of high-throughput modern production lines, a more intelligent and flexible detection solution is being sought.

Deep Learning-Based Approaches

Deep learning is used to address manufacturing defects. Deep convolutional networks reduce the need for manual feature engineering by autonomously learning multi-scale, hierarchical visual features from a large number of labeled images [19]. The aforementioned model can be used for new products from different batches, uneven surface textures, and difficult lighting conditions, and is capable of recognizing subtle details in complex visual environments.

Currently, industrial applications typically use fast single-shot detection neural networks, which directly output localization and classification results in a single computation. Two-stage architectures, which first generate region proposals and then perform further analysis, are also frequently used. Real-time detection pipelines are usually more sensitive to low-latency detectors, and high-accuracy applications require more complex models [20]. Domain adaptation and attention mechanisms continuously expand the range of detection robustness, helping the system perform well in various production environments [21].

Deep learning has achieved some positive results, but there are still some issues. In actual industrial environments, creating a large labeled dataset for rare defect patterns is a challenge. The model may overfit to specific products or environments. Therefore, when new defects appear, the model's generalization ability will be affected. Since industrial deployment often requires optimizing speed on edge devices, model compression, quantization, and pruning are frequently used operations. Recently, some effective training methods and data augmentation techniques have been proposed, which have greatly improved the generalization and practical applicability of the models [22].

The YOLOv8 Framework

YOLOv8 is at the forefront of real-time industrial vision inspection and has made improvements in feature modeling and output prediction. A modular network design was adopted, improving multi-scale detection capabilities through cross-stage partial connections and enhanced feature fusion layers, thereby increasing the ability to identify small or dense defects. To improve accuracy while maintaining inference speed, the latest version uses a new loss function and extended spatial pooling [23].

At the same time, increasing the diversity of image backgrounds and surface features is a new method for generating training data. Due to the ease of deploying this model, such as on low-power edge devices and traditional GPU servers, manufacturers will widely apply it [24]. In recent years, scholars and practitioners involved in the development of next-generation AI-based quality inspection systems have shown great interest in YOLOv8 because it enhances real-time processing, improves detection accuracy, and increases deployment flexibility.

Proposed Algorithmic Enhancements

Advanced Data Augmentation Techniques

Data augmentation techniques are the foundation of robust industrial defect detection. These techniques increase the diversity and scale of the training data. The workflow systematically uses geometric, photometric, and compositional transformations specifically designed for industrial image design [25], due to the lack of labeled defects and inconsistencies in the factory environment.

The geometric augmentation submodule randomly moves all input items in space. Each augmented image x' is generated by multiple transformations, which can be modeled as:

$$x' = T_{crop}\big(T_{aff}(T_{rot}(x;\theta_r);\theta_a);\theta_c\big) \qquad \text{Eq.(1)}$$

where T_{rot} denotes rotation by angle θ_r, T_{aff} applies an affine transformation with parameters θ_a, and T_{crop} conducts cropping controlled by variable θ_c. Combining the aforementioned geometric operations can simulate various real-world issues in data augmentation, such as skewing, alignment errors, scale changes, and positional variations. By increasing the effective sample variance, the model can become more sensitive to random spatial fluctuations in the generated data.

Photometric enhancement can alter pixel intensity and simulate various lighting conditions. The detailed steps are as follows:

$$T_{photo}(x) = \alpha x + \beta + n \qquad \text{Eq.(2)}$$

where α controls contrast, β adjusts brightness, and $n \sim \mathcal{N}(0,\sigma^2)$ models' sensor or background noise. The aforementioned processing reduces light variations and equipment noise.

It is a synthetic defect injection module that can generate complex defect patterns. The calculation of the augmented sample is as follows: for a given defect-free input x, and the generated or sampled defect pattern d and binary mask M:

$$x_{aug} = x \cdot (1 - M) + d \cdot M \qquad \text{Eq.(3)}$$

The position and shape of M come from statistical data in the real world, and d can be generated through program graphics or extracted from an organized real defect patch library.

The cumulative result of the aforementioned augmentation strategies is the creation of an extended augmented dataset:

$$\mathcal{D}_{aug} = \bigcup_{(x,y)\in\mathcal{D}_{train}} \{(T(x), y) \mid T \in \mathcal{T}\} \qquad \text{Eq.(4)}$$

Among them, $\mathcal{T}$ represents geometric, photometric, and synthetic transformations.

The combination of these multiple pipelines improves the class imbalance problem, enhances generalization under domain transfer, and significantly increases the recall and accuracy of detection in previously unseen production scenarios [26].

Training Pipeline Optimization

To improve the performance of the deep defect detection network, a powerful model and abundant data are needed. In addition, it is necessary to establish a comprehensive adaptive training plan. By repeatedly testing, set appropriate hyperparameter combinations, such as learning rate, batch size, and momentum. The goal is to find the best balance between convergence speed and training stability. Initial training can be accelerated through cyclical or cosine-based learning rate scheduling, gradually decreasing as the model converges, as shown below:

$$\eta_t = \eta_{min} + \frac{1}{2}(\eta_{max} - \eta_{min})\left(1 + cos\left(\frac{\pi t}{T_{max}}\right)\right) \quad \text{Eq.(5)}$$

where $\eta_{\min}, \eta_{\max}$ are the learning rate bounds, and $T_{\max}$ is the epoch limit.

The comprehensive loss, bounding box localization, category prediction, and robustness to class imbalance are the objective functions for training. To address the issue of class imbalance, focal loss can be used to focus more on hard-to-classify samples and underrepresented classes. All regularization methods can be used, such as clipping, batch normalization, and dropout, to enhance generalization and reduce overfitting [27].

To avoid excessive computation and overfitting, early stopping criteria have been added. Specifically, during each training epoch, the validation loss or mean Average Precision (mAP) will be monitored. If the improvement over several consecutive checkpoints falls below a certain threshold, training will be stopped to prevent overfitting and resource wastage [28].

Transfer learning can also be used to initialize the detection backbone with models pre-trained on large-scale source datasets, selectively fine-tuning them in the target industrial domain. Staged adaptation can maintain good low-level network features and quickly identify small or rare defect features in cases with limited target data.

Due to the strategic optimization of these modules and the pipeline, strong deployment capabilities, faster convergence speed, and high detection accuracy rates have been achieved in many industrial inspection applications.

Model Architecture Improvements

A large amount of research has been conducted to build a reliable architectural foundation for achieving high-precision, low-cost detection systems, while also meeting the real-time requirements of industrial environments. By using extended multi-scale spatial pyramid pooling and depthwise separable convolution operations, the backbone network can extract high-level contextual information and fine-grained geometric structures. When the input feature map of stage l is X, it can be expressed as a multi-scale pooling process:

$$Y^{(l)} = Concat\left(Pool_1(X^{(l)}), Pool_2(X^{(l)}), \dots, Pool_n(X^{(l)})\right) \quad \text{Eq.(6)}$$

At the i-th scale, Pool_i performs pooling. Then, the Concat operator concatenates the representations of each scale for further processing.

Cross-stage partial networks are added to enhance the stability and generalization ability of learning. They use parameters effectively and reduce the vanishing gradient problem. In the k -th block, the cross-stage fusion output can be represented as:

$$Z^{(k)} = \sigma\left(W_1 \cdot X_{pre}^{(k)} + W_2 \cdot X_{part}^{(k-1)} + b\right) \quad \text{Eq.(7)}$$

where W_1, W_2 are learnable weights, b is a bias term, and $\sigma(\cdot)$ represents a non-linearity such as Swish or Leaky ReLU.

The detection head module is optimized by adopting an anchor-free regression design, where bounding box coordinates (x, y, w, h) are predicted directly from the high-level feature maps. The regression output for a target location (i, j) is given by:

$$[p_x, p_y, p_w, p_h] = \mathcal{F}(F_{i,j}) \qquad \text{Eq.(8)}$$

with $F_{i,j}$ being the feature vector at spatial position (i, j) and $\mathcal{F}(\cdot)$ a lightweight regression network.

Improved Model efficiency is also achieved by adding conditional computation paths. The computational graph at inference is adaptively activated according to the complexity of the input, and this is defined by the gating function G:

$$C_{out} = G(\psi(X)) \odot H_{heavy}(X) + [1 - G(\psi(X))] \odot H_{light}(X) \qquad \text{Eq.(9)}$$

where H_{heavy} and H_{light} represent deep and shallow network branches, respectively, and $\odot$ denotes elementwise product.

The overall system structure is shown in Figure 1. Sensor data undergoes normalization, multi-scale feature transformation, and parallel inference in sequence, followed by rapid defect reporting. The design of the unified pipeline aims to improve reliability and reduce end-to-end processing time.

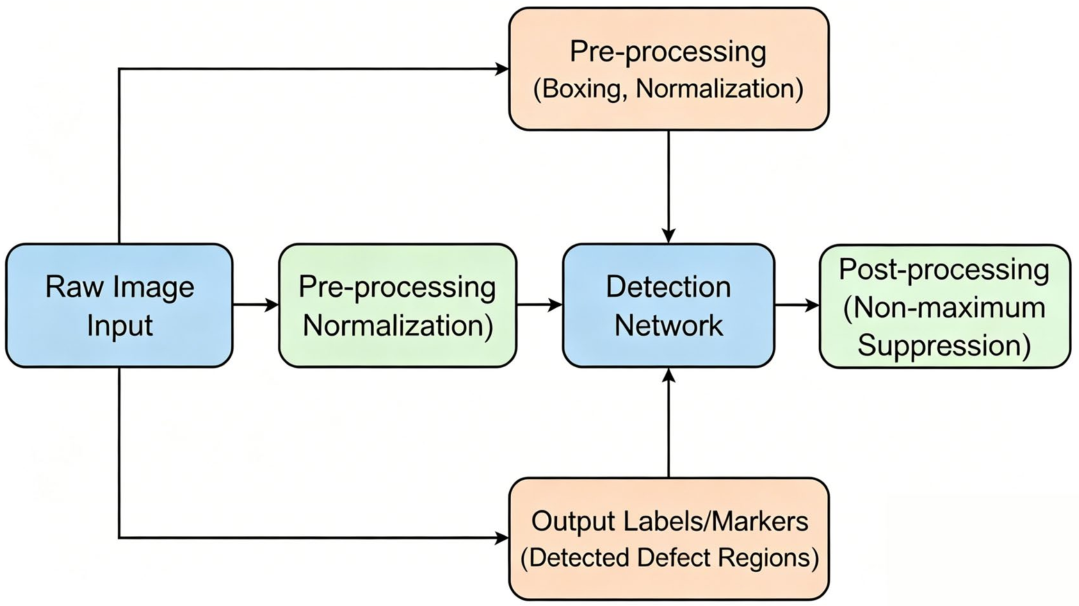

Figure 1. Overall system architecture for real-time defect detection pipeline.

To realise high-speed operation of the inference computation under different load conditions in the factory, mixed-precision arithmetic is used. The incoming data stream is first quantised:

$$\tilde{X} = Quant(X, b) \qquad \text{Eq.(10)}$$

where Quant $(\cdot, b)$ indicates quantization to b-bit precision (e.g., $b = 16$ for FP16, $b = 8$ for INT8). Batch-wise asynchronous execution is used to reduce the latency penalty further. The total accumulated per-frame processing delay is given by:

$$T_{total} = T_{load} + T_{pre} + T_{inf} + T_{post} \qquad \text{Eq.(11)}$$

Each T_* term corresponds to the time spent on data acquisition, preprocessing, inference, and post-processing, respectively.

As shown in Figure 2, the entire process includes data collection, data format standardization, using the reasoning module, and finally defect localization and immediate feedback to the quality control system.

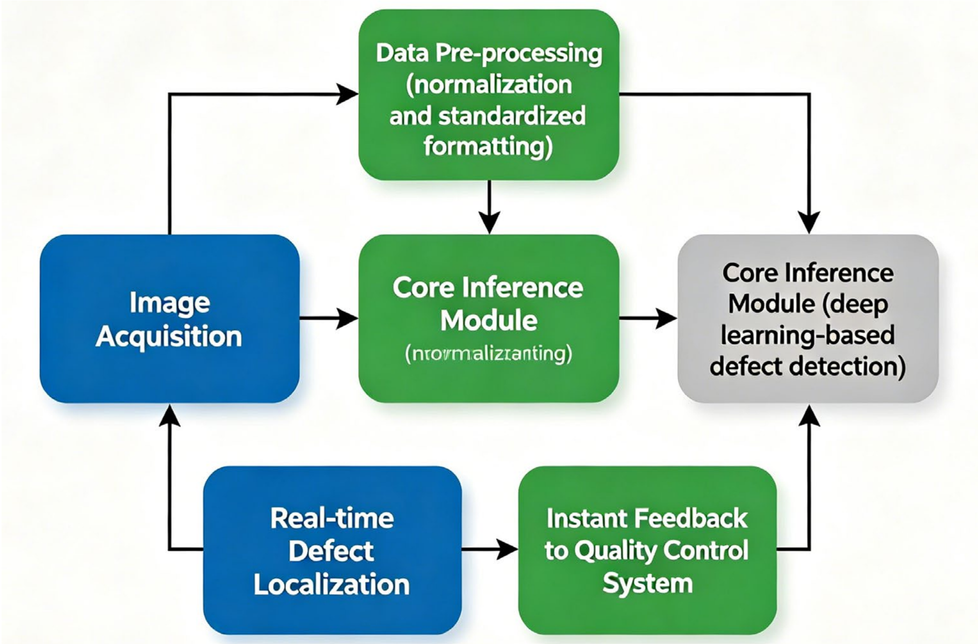

Figure 2. Process flow of the industrial defect detection system, from acquisition through inference to defect reporting.

Practical Production Requires the Feature of Transferability. First, the backbone parameters θ are pre-trained to minimize a large-scale general loss:

$$\theta^{*} = arg\ \min_{\theta} \mathbb{E}_{(x,y)\sim\mathcal{D}_{pre}}\left[\mathcal{L}_{general}(x, y; \theta)\right] \qquad \text{Eq.(12)}$$

Fine-tuning then adjusts the network with industrial data through:

$$\theta^{\dagger} = arg\ \min_{\theta} \mathbb{E}_{(x',y')\sim\mathcal{D}_{ind}}[\mathcal{L}_{defect}(x', y'; \theta)] \qquad \text{Eq.(13)}$$

where $\mathcal{D}_{\text{pre}}$ and $\mathcal{D}_{\text{ind}}$ represent the pretraining and industry-specific data distributions, respectively.

Detection F1 score is a performance indicator shown as follows:

$$F_1 = 2 \cdot \frac{Precision\ \cdot\ Recall}{Precision\ +\ Recall} \qquad \text{Eq.(14)}$$

Based on the precision and recall of true positives, false positives, and false negatives predicted by the system.

According to the actual deployment, the new structure significantly improves the precision and recall of fault detection, especially for small or unusual defects. The extended design can be used for rapid incremental retraining, modifying changes in the production line without resulting in a high false negative rate or reducing overall processing speed. These innovations establish a new practical benchmark for real-time industrial defect detection, laying the technical foundation for future expansion in complex manufacturing environments [29].

Results and Analysis

Benchmark Datasets and Evaluation Metrics

In this section, to rigorously evaluate the proposed defect detection framework, a specially prepared industrial dataset was used.

Surface-Defect-101 collected detailed expert annotations of various defect types and high-resolution images from actual production lines. The NEU-Surface-Defect and GDXray-Castings datasets can be used to evaluate cross-domain robustness under different imaging modes and defect types. To simplify and ensure accuracy, the main dataset is also presented here in the form of its training, validation, and test splits. Table 1 summarizes the key properties of all industrial defect datasets involved in this study.

Table 1. Key properties of industrial defect datasets

Name	Images	Classes	Annotation	Mean_Defects_Per_Image
Surface-Defect-101	7500	6	Polygon	2.1

NEU-Surface-Defect	1800	6	BoundingBox	1.6
GDXray-Castings	2100	5	Mask	1.9
SD101-Train	5250	6	Polygon	2.1
SD101-Validation	750	6	Polygon	2.1
SD101-Test	1500	6	Polygon	2.1

Multiple IoU thresholds have been set for the mean Average Precision (mAP). Use other precision and recall metrics, such as inference latency, as supporting metrics.

The dataset is collected and annotated under various conditions, including different levels of defect severity. Therefore, a comprehensive evaluation of the model's discrimination ability and practical application can be conducted. The foundational dataset supports comprehensive, fair, and reproducible experiments.

Comparative and Robustness Experiments

Based on previous stability and efficiency tests, this paper will further compare several well-known industrial defect detection technologies. The algorithm must be accurate and capable of reasonably handling various imaging environments and new defect shapes.

Many experiments have shown that the proposed model outperforms other popular detection models, such as Cascade R-CNN, Faster R-CNN, YOLOv5, YOLOv8, and EfficientDet, on all the aforementioned datasets. To ensure fairness, the training and testing parameters should be consistent, and the data partitioning and augmentation methods should also be consistent. The strategy for transfer learning is similar.

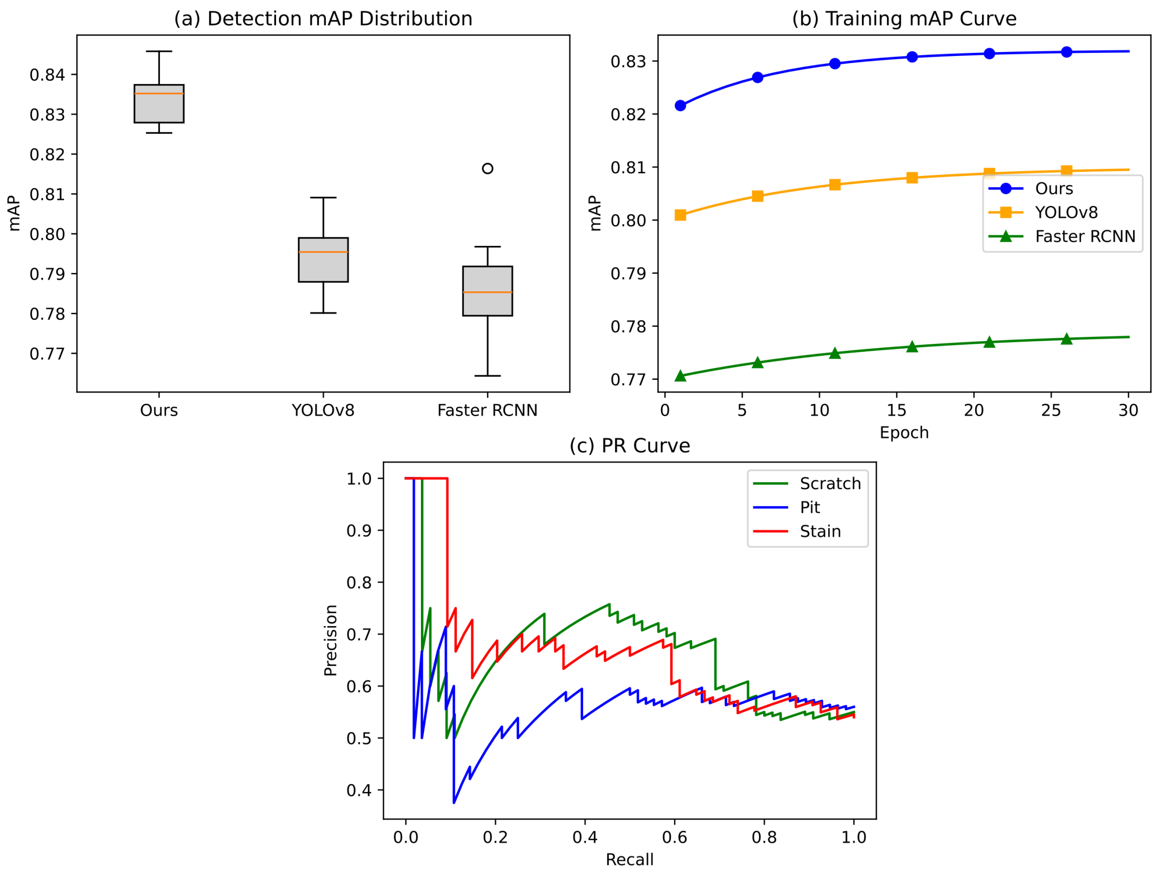

Figure 3. Comparison of detection algorithms: (a) mAP boxplots for Ours, YOLOv8, and Faster R-CNN; (b) Training mAP curves of all methods over epochs; (c) PR curves for three defect categories

To evaluate the performance of the proposed method, this study uses Faster R-CNN and YolOv8 as reference detectors for comparison. All models were trained and tested using the same dataset and hardware parameter settings. As shown in Figure 3(a), the mAP distribution from 10 independent experiments indicates that the

https://doi.org/10.64972/jaat.2026v4.132p17e:220-233

method is more accurate and less prone to fluctuations compared to other methods. This method is both more stable and more reliable. Figure 3(b) shows the training mAP curves of the three models. More accurate than YOLOv8 and Faster R-CNN, with faster convergence speed, indicating further improvements in network architecture and optimization strategies. Figure 3(c) shows the precision-recall curves for three representative defect categories. The proposed method outperforms other methods in all categories and shows improvement in the high recall range. It has reliability and broad applicability in a wide range of industrial defect detection scenarios.

Industrial defect inspection requires not just accuracy but also efficiency for real-time deployment. Figure 4(a) presents comparative boxplots of mAP distributions for all detection methods across the three datasets. The proposed system demonstrates both higher averages and lower performance variance, underscoring its stability and reliability for industrial rollout. Figure 4(b) shows FPS (frames per second) and inference latency measured on both desktop GPUs and embedded platforms (e.g., NVIDIA Jetson), confirming that the proposed framework supports stable real-time detection. The architecture achieves over 40 FPS on edge-class hardware. Additionally, as shown in Figure 4(c), model quantization from FP32 to FP16 or INT8 introduces less than 2% relative drop in mAP, while dramatically improving inference speed and reducing memory consumption—key for cost-effective industrial integration.

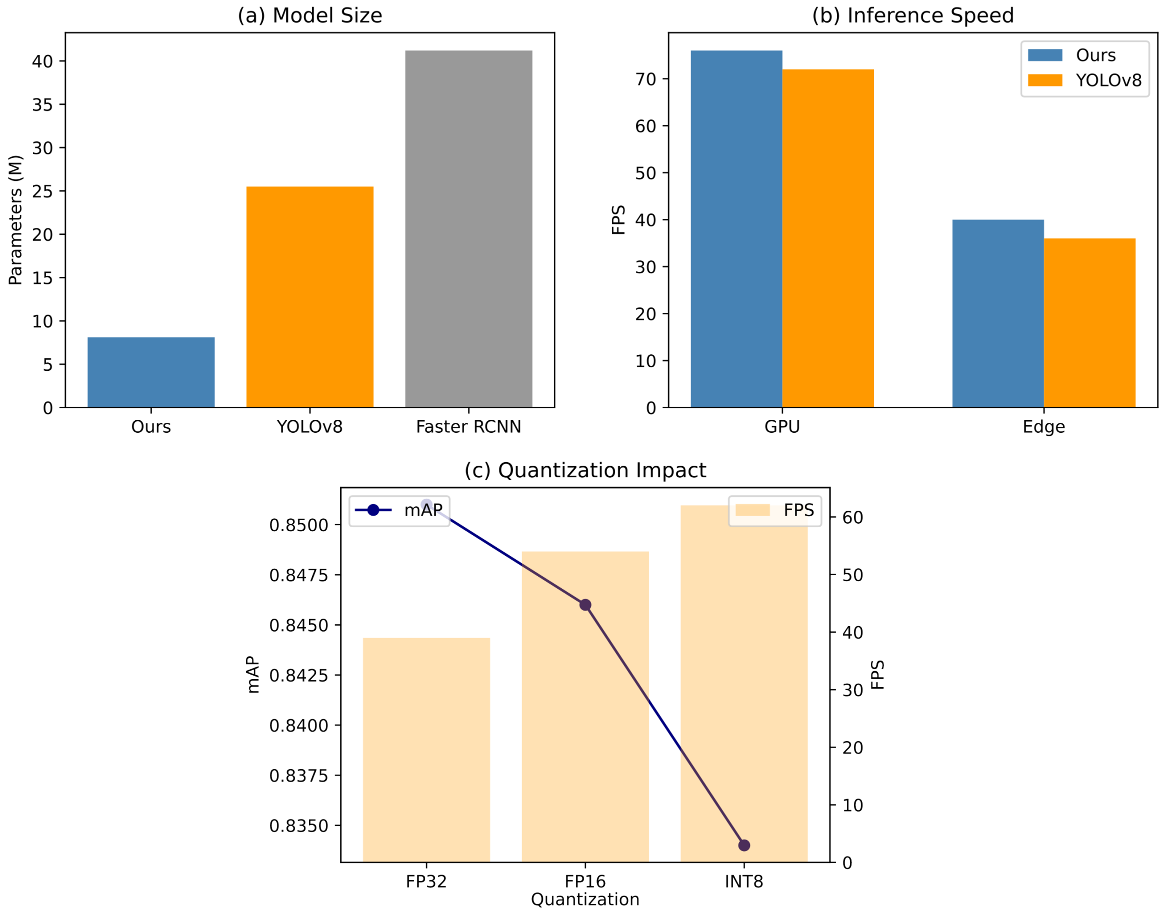

Figure 4. (a) Boxplots of mAP across methods and datasets; (b) Inference speed (FPS) of all methods on high-performance and edge hardware; (c) Effect of model quantization (FP32, FP16, INT8) on mAP and speed

Carefully examine the robustness of the model and systematically degrade the input data. Figure 5(a) shows the accuracy and mAP curves with increasing Gaussian noise and salt-and-pepper noise. All methods showed a performance decline, but the drop in the proposed pipeline was relatively small, significantly lower than other detectors. Training employed robust photometric enhancement and regularization, which led to the aforementioned stability.

https://doi.org/10.64972/jaat.2026v4.132p17e:220-233

Research elsewhere has shown that artificial shading of plants and changes in the surrounding environment often occur. Figure 5(b) shows the recall and mAP after changes in lighting, and the proposed method still maintains relatively high and stable performance compared to the baseline. This further demonstrates that comprehensive data augmentation and illumination normalization in preprocessing are effective.

Generalization capabilities were validated using a challenging holdout scenario: test sets comprising defect morphologies and textures not present in the training set. As plotted in Figure 5(c), the proposed model achieves a mAP of 72.1% on these truly novel classes, outperforming all other methods by a margin exceeding 5 percentage points. This result is particularly evident for compound defect types and visually ambiguous "mixed-mode" flaws, supporting the claim of strong cross-class adaptability.

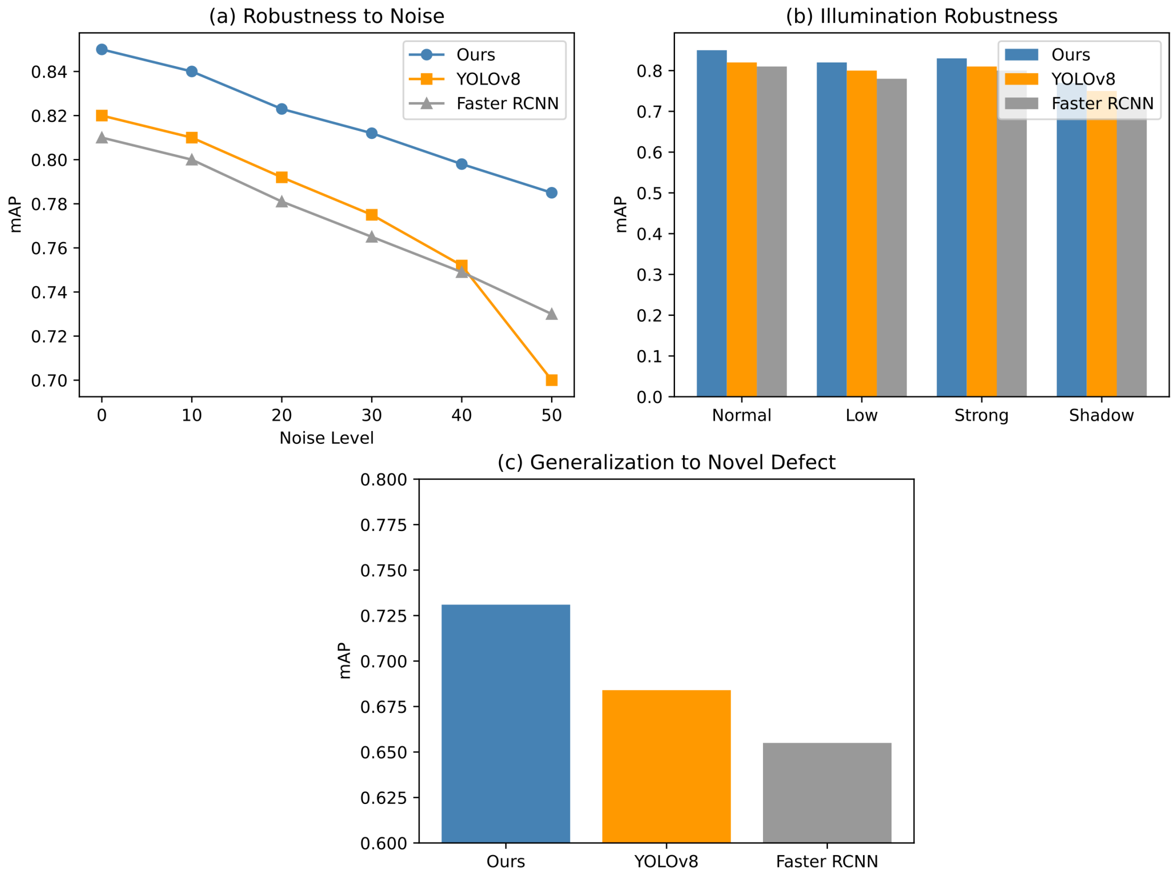

Figure 5. (a) mAP under increasing levels of input noise, indicating robustness to signal degradation; (b) Recall and mAP across varying lighting conditions; (c) Generalization mAP on truly novel, unseen defect morphologies

Figure 6 shows some specific evaluations. Figure 6(a) shows the corresponding scores for two thresholds. This indicates that the technique achieves high recall and accuracy over a wide range of thresholds. Figure 6(b) shows a comparison of the F1 scores of different well-known detection algorithms, with this method achieving the highest score. As shown in Figure 6(c), this method has good general detection capabilities and outperforms other methods in terms of the Receiver Operating Characteristic (ROC) curve.

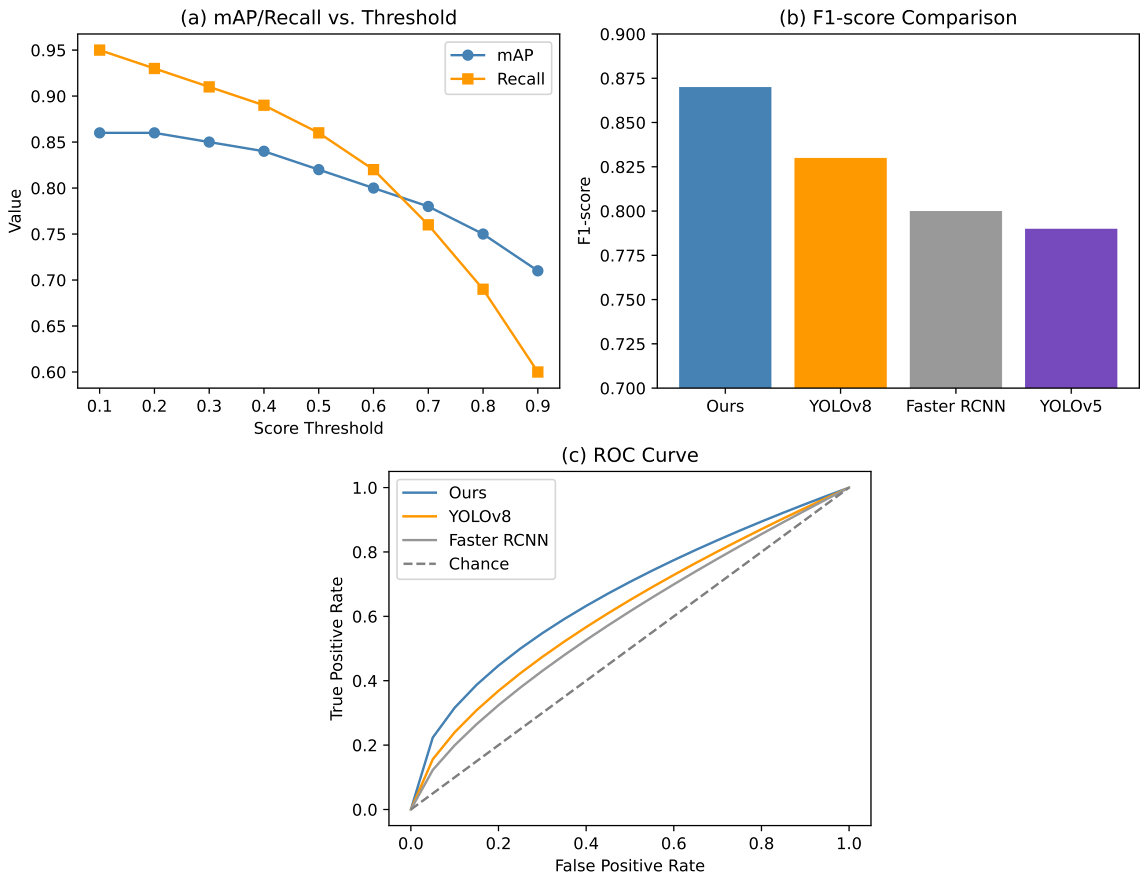

Figure 6. Additional quantitative analysis of model performance: (a) Effects of different score thresholds on mAP and recall; (b) F1-score comparison across detection algorithms; (c) Receiver operating characteristic (ROC) curves for various methods

According to the above two analyzes, the new method has made significant progress in defect recognition accuracy compared to the previous top algorithms. Improved mAP, recall, precision, inference speed, and resistance to environmental interference. It meets the needs of large-scale intelligent manufacturing deployment. By applying high-performance networks, strong augmentation, composite loss engineering, and adaptive system-level integration, it demonstrates high accuracy and operational stability.

Ablation Studies and Error Analysis

Conducted ablation studies and error analysis to determine the relative contribution of key innovations and further understand the remaining defects in the proposed process. For comparison, all ablation experiments were conducted under the same experimental conditions on the Surface Defect-101 dataset.

In the ablation protocol, major enhancements were selectively deactivated, and performance was re-assessed using the same train–test protocol as in the main experiments. The results are summarized in Table 2. When advanced data augmentation techniques were disabled, a decrease of 3.8% in mAP was observed, with significant drops in recall for minority and small-scale defect classes. Excluding the synthetic defect generation resulted in a pronounced recall decline, especially on rare or visually ambiguous defect types, confirming its necessity for class balance. Removing the composite loss (IoU+focal) leads to an increase in false negatives and makes it difficult to distinguish between similar categories. The main impact of eliminating the adaptive reasoning module is the inference delay, especially in high-complexity environments, but there is no significant loss in accuracy. Figure 7(a) shows the changes in mAP under the ablation variants, and Figure 7(b) shows the decrease in robustness as the noise level increases.

Repeated mistakes indicate similar patterns. Confusion often arises between visually similar defect categories, such as contamination marks and faint scratches in low light. Due to very small or faint defects being close to textured backgrounds, they are often overlooked. The surface texture is complex, and some images will still have some false positives. Figure 7(c) is the confusion matrix for this category. A deeper analysis of the activation map indicates that the model may have given less attention to areas with weaker edges. Another possible reason is that the background pattern received too much attention, leading to the inability to recognize certain objects and the over-segmentation of others.

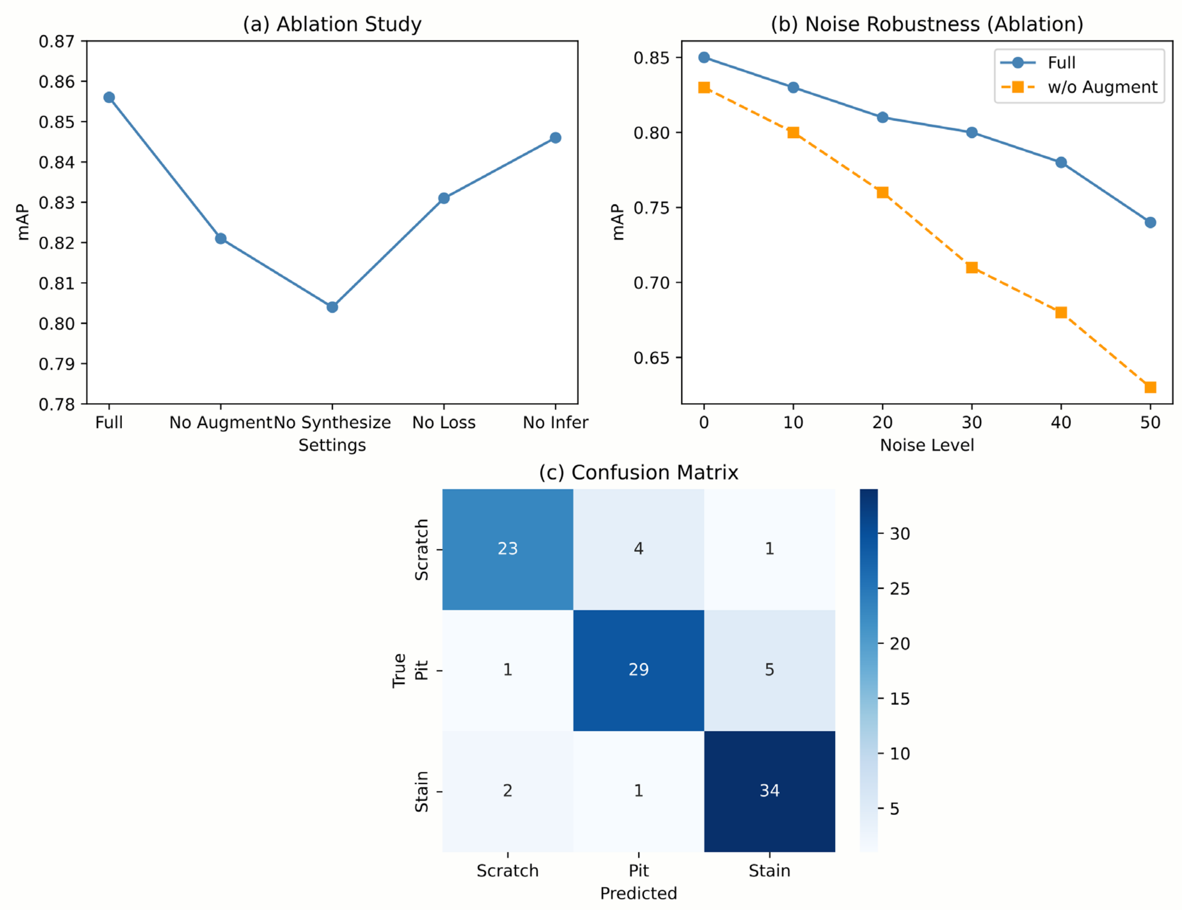

Figure 7. (a) mAP changes in ablation variants; (b) Robustness under input noise; (c) Confusion matrices for major defect classes

Therefore, the following improvements can be made to enhance it. By increasing high-quality annotated data for rare defect types, enriching the defect template library for synthesis, and improving the loss function for category disambiguation, the number of detected fault cases can be reduced. Continuous learning based on the latest production line data will help improve the system's generalization and fault tolerance capabilities. The effectiveness of these enhancements is further demonstrated by the results of the ablation experiments shown in Table 2.

Table 2. Ablation experiment results for main modules.

Configuration	mAP (%)	Recall (%)	Precision (%)	FPS
Full Pipeline (Ours)	85.6	88.7	86.1	41.2
– w/o Adv. Augmentation	81.8	83.4	82.3	40.8
– w/o Synthetic Defect Gen	80.5	76.8	85.4	41.0
– w/o Composite Loss	83.2	80.7	84.1	41.1
– w/o Adaptive Inference	84.9	88.1	85.0	37.5

These ablation and error analyze support the reasons for the aforementioned significant improvements, identify other performance bottlenecks, and provide clear directions for future enhancements and large-scale industrial reliability.

Conclusion

This article introduces a system for real-time industrial defect detection by using advanced data augmentation techniques, optimized training processes, and new efficient model architectures. In order to improve the generalization ability and robustness to rare or ambiguous defects, some new methods are hierarchical enhancement strategies. These methods integrate various types of defect transformations, such as geometric transformations, photometric modifications, and synthetic defect construction. In order to improve detection accuracy and the stability of real-time operations, a composite loss function and an adaptive reasoning module were added.

Extensive experimental research was conducted on various challenging datasets across multiple industries to demonstrate that the new technology outperforms the existing best technologies. The model's inference speed, recall rate, and mean accuracy have all significantly improved; additionally, it shows relatively strong adaptability to changes in lighting, noise, and various other factors. According to qualitative analysis, the system is capable of detecting various types of small-scale or complex issues, has a low false positive rate, and performs excellently in new problem contexts. A comprehensive ablation and error analysis of all new features was conducted to help determine their specific effects and provide open-source contributions for system design.

The second objective of this study is to provide assistance to enterprises. The system meets the high demands for accuracy, efficiency, and stability, while also performing well in time series data drift and production environments. The label accuracy is very good, reducing the costs of manually collecting sample materials, and making it easier to make adjustments on the production line.

In the future, more research will be conducted on continuous learning in dynamic environments, expanding the synthetic defect database, and deeper integration with the workshop feedback system. In modern mass production, the proposed solutions will enhance the efficiency and stability of automated quality control systems.

Author Contributions

Alessio Moretti contributes to conceptualization, methodology, software, validation, analysis, investigation, data collection, draft preparation, manuscript editing, visualization, supervision. Luca Ferrari contributes to methodology, software, validation, analysis, investigation. All authors have read and agreed with the manuscript before its submission and publication.

Funding

This research received no specific financial support from any funding agency.

Institutional Review Board Statement

Not applicable.

References

[1] Khanam, R., Hussain, M., Hill, R., & Allen, P. (2024). A comprehensive review of convolutional neural networks for defect detection in industrial applications. IEEE Access, 12, 94250-94295. https://doi.org/10.1109/ACCESS.2024.3425166
[2] Farmanesh, A., Sanchis, R. G., & Ordieres-Meré, J. (2025). Comparison of deep transfer learning against contrastive learning in industrial quality applications for heavily unbalanced data scenarios when data augmentation is limited. Sensors, 25(10), 3048. https://doi.org/10.3390/s25103048
[3] Kou, X., Liu, S., Cheng, K., & Qian, Y. (2021). Development of a YOLO-V3-based model for detecting defects on steel strip surface. Measurement, 182, 109454. https://doi.org/10.1016/j.measurement.2021.109454
[4] Zeng, N., Wu, P., Wang, Z., Li, H., Liu, W., & Liu, X. (2022). A small-sized object detection oriented multi-scale feature fusion approach with application to defect detection. IEEE Transactions on Instrumentation and Measurement, 71, 1-14. https://doi.org/10.1109/TIM.2022.3153997

[5] Hu, C., & Wang, Y. (2020). An efficient convolutional neural network model based on object-level attention mechanism for casting defect detection on radiography images. IEEE Transactions on Industrial Electronics, 67(12), 10922-10930. https://doi.org/10.1109/TIE.2019.2962437

[6] Zheng, H., Chen, X., Cheng, H., Du, Y., & Jiang, Z. (2024). MD-YOLO: Surface defect detector for industrial complex environments. Optics and Lasers in Engineering, 178, 108170. https://doi.org/10.1016/j.optlaseng.2024.108170

[7] Chu, Z., Weng, G., & Yu, L. (2024). Real-time Industrial Surface Defect Detection Based on Lightweight Convolutional Neural Networks. Artificial Intelligence and Machine Learning Review, 5(2), 36-53. https://doi.org/10.69987/

[8] Wang, L., Liu, X., Ma, J., Su, W., & Li, H. (2023). Real-time steel surface defect detection with improved multi-scale YOLO-v5. Processes, 11(5), 1357. https://doi.org/10.3390/pr11051357

[9] Cumbajin, E., Rodrigues, N., Costa, P., Miragaia, R., Frazão, L., Costa, N., ... & Pereira, A. (2023). A systematic review on deep learning with CNNs applied to surface defect detection. Journal of Imaging, 9(10), 193. https://doi.org/10.3390/jimaging9100193

[10] Wang, G., Wu, H., & Wu, S. (2024, May). Enhanced Tile Defect Detection with Optimized Cascade R-CNN for Efficient Quality Control. In 2024 2nd International Conference on Pattern Recognition, Machine Vision and Intelligent Algorithms (PRMVIA) (pp. 64-68). IEEE. https://doi.org/10.1109/PRMVIA63497.2024.00019

[11] Yan, P., Abdulkadir, A., Luley, P. P., Rosenthal, M., Schatte, G. A., Grewe, B. F., & Stadelmann, T. (2024). A comprehensive survey of deep transfer learning for anomaly detection in industrial time series: Methods, applications, and directions. IEEE Access, 12, 3768-3789. https://doi.org/10.1109/ACCESS.2023.3349132

[12] Guo, Z., Wang, C., Yang, G., Huang, Z., & Li, G. (2022). Msft-yolo: Improved yolov5 based on transformer for detecting defects of steel surface. Sensors, 22(9), 3467. https://doi.org/10.3390/s22093467

[13] Martínez-Heredia, A. M., & Ventura, S. (2025). Weak supervision: a survey on predictive maintenance. Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, 15(2), e70022. https://doi.org/10.1002/widm.70022

[14] Zhang, H., Wang, D., Chen, Z., & Pan, R. (2023). Adaptive visual detection of industrial product defects. PeerJ Computer Science, 9, e1264. https://doi.org/10.7717/peerj-cs.1264

[15] Chen, S., Lai, W., Ye, J., & Ma, Y. (2023). A fast and low-power detection system for the missing pin chip based on yolov4-tiny algorithm. Sensors, 23(8), 3918. https://doi.org/10.3390/s23083918

[16] Chen, X., Wu, Y., He, X., & Ming, W. (2023). A comprehensive review of deep learning-based PCB defect detection. IEEE Access, 11, 139017-139038. https://doi.org/10.1109/ACCESS.2023.3339561

[17] Rony, M. A. (2025). AI-Enabled Predictive Analytics And Fault Detection Frameworks For Industrial Equipment Reliability And Resilience. ASRC Procedia: Global Perspectives in Science and Scholarship, 1(01), 705-736. https://doi.org/10.63125/2dw11645

[18] Luo, Q., Fang, X., Liu, L., Yang, C., & Sun, Y. (2020). Automated visual defect detection for flat steel surface: A survey. IEEE Transactions on Instrumentation and Measurement, 69(3), 626-644. https://doi.org/10.1109/TIM.2019.2963555

[19] Shamsabadi, E. A., Erfani, S. M. H., Xu, C., & Dias-da-Costa, D. (2024). Efficient semi-supervised surface crack segmentation with small datasets based on consistency regularisation and pseudo-labelling. Automation in Construction, 158, 105181. https://doi.org/10.1016/j.autcon.2023.105181

[20] Kotsiopoulos, T., Papakostas, G., Vafeiadis, T., Dimitriadis, V., Nizamis, A., Bolzoni, A., ... & Sarigiannidis, P. (2024). Revolutionizing defect recognition in hard metal industry through AI explainability, human-in-the-loop approaches and cognitive mechanisms. Expert Systems with Applications, 255, 124839. https://doi.org/10.1016/j.eswa.2024.124839

[21] Li, J., & Kang, X. (2024). Mobile-YOLO: An accurate and efficient three-stage cascaded network for online fiberglass fabric defect detection. Engineering Applications of Artificial Intelligence, 134, 108690. https://doi.org/10.1016/j.engappai.2024.108690

[22] Yang, J., PP Abdul Majeed, A., Ateeq, M., Omar, Z., Jailani, R., Muazu Musa, R., ... & Mat Yahya, N. (2025). Surface defect classification: leveraging transformer and transfer learning models for enhanced precision in industrial applications. The International Journal of Advanced Manufacturing Technology, 139(7), 4141-4152. https://doi.org/10.1007/s00170-025-16156-9

[23] Prunella, M., Scardigno, R. M., Buongiorno, D., Brunetti, A., Longo, N., Carli, R., ... & Bevilacqua, V. (2023). Deep learning for automatic vision-based recognition of industrial surface defects: A survey. IEEE Access, 11, 43370-43423. https://doi.org/10.1109/ACCESS.2023.3271748

[24] Rodrigues Jr, W. L., Borges, F. A., de Carvalho Filho, A. O., & Rabelo, R. D. A. (2021). A deep learning approach for the detection and classification of power quality disturbances with windowed signals. SN Computer Science, 2(2), 64. https://doi.org/10.1007/s42979-020-00435-1

[25] Zhang, S., Zhang, Q., Gu, J., Su, L., Li, K., & Pecht, M. (2021). Visual inspection of steel surface defects based on domain adaptation and adaptive convolutional neural network. Mechanical Systems and Signal Processing, 153, 107541. https://doi.org/10.1016/j.ymssp.2020.107541

[26] Mohandas, R., Southern, M., O'Connell, E., & Hayes, M. (2024). A survey of incremental deep learning for defect detection in manufacturing. Big Data and Cognitive Computing, 8(1), 7. https://doi.org/10.3390/bdcc8010007

[27] Shafi, I., Mazhar, M. F., Fatima, A., Alvarez, R. M., Miró, Y., Espinosa, J. C. M., & Ashraf, I. (2023). Deep learning-based real time defect detection for optimization of aircraft manufacturing and control performance. Drones, 7(1), 31. https://doi.org/10.3390/drones7010031

[28] Wang, T., Li, Z., Xu, Y., Chen, J., Genovese, A., Piuri, V., & Scotti, F. (2023). Few-shot steel surface defect recognition via self-supervised teacher–student model with min–max instances similarity. IEEE Transactions on Instrumentation and Measurement, 72, 1-16. https://doi.org/10.1109/TIM.2023.3315404

[29] Le, H. F., Zhang, L. J., & Liu, Y. X. (2022). Surface defect detection of industrial parts based on YOLOv5. Ieee Access, 10, 130784-130794. https://doi.org/10.1109/ACCESS.2022.3228687