Intelligent Methods in Engineering Sciences 5(2): 55-68, 2026

IMIENS

INTELLIGENT METHODS
IN ENGINEERING SCIENCES

https://www.imiens.org

International
Open Access

August, 2026

e-ISSN 2979-9236

Research Article

https://doi.org/10.58190/imiens.2026.175

YOLO12 and YOLO13 for Steel Surface Defect Detection: Experimental Evaluation

Adem DILBAZ [a,*]

[a] *Sahin Tanker Limited Company, Asagipinarbasi Organized Industrial Zone, 521 Street 7, 42250, Konya, Türkiye*

ARTICLE INFO

Article history:
Received 22 July 2026
Accepted 15 August 2026

Keywords:
Deep learning,
Steel surface defect,
Object Detection,
YOLO12, YOLO13

ABSTRACT

Intelligent inspection and defect detection of steel surfaces are essential for ensuring product quality, minimizing production losses, and supporting quality control in modern manufacturing systems. In particular, the identification of defects on steel plates is an important step, as surface imperfections can significantly affect the mechanical performance and commercial value of the final product. This study presents a comparative evaluation of YOLO12 and YOLO13 architectures for steel surface defect detection using the NEU-DET dataset under different validation ratios and input image resolutions. Four models (YOLO12-S, YOLO12-L, YOLO13-S, and YOLO13-L) were evaluated under three experimental settings using standard object detection metrics and three different random seeds. Experimental results showed that YOLO13-S consistently demonstrated superior overall detection performance, reaching a best single-run mAP@50 value of 0.753 with seed 42 and a precision value of 0.848 with seed 44. In addition, it achieved the highest single-run mAP@50–95 value of 0.449, with seed 43. In contrast, YOLO12-L achieved the highest single-run recall (0.749), with seed 42, indicating its stronger capability for detecting positive defect instances. Overall, the results demonstrate that YOLO13-S provides the best balance between detection accuracy and computational efficiency, while YOLO12-L offers superior recall performance under the evaluated conditions.

1. INTRODUCTION

Intelligent inspection and defect detection of steel surfaces have become increasingly important in modern manufacturing processes to ensure product quality, maintain mechanical performance, and improve customer satisfaction. In particular, hot-rolled steel plates are highly susceptible to surface defects such as cracks, inclusions, scratches, and rolled-in scale. These defects may negatively affect structural integrity, reduce material reliability, and decrease the commercial value of manufactured products. Traditional inspection methods generally rely on manual visual examination or rule-based image analysis techniques, which are often limited in terms of accuracy, scalability, and efficiency under complex industrial conditions.

In recent years, artificial intelligence (AI) and deep learning-based approaches have enabled the development of automated defect inspection systems with improved accuracy and processing capability compared with conventional methods. Among these approaches, You Only Look Once (YOLO)-based object detection models have attracted considerable attention due to their strong localization and classification capabilities. Their ability to process image datasets using various quantitative evaluation metrics makes them particularly suitable for industrial inspection applications, especially in scenarios involving limited-scale datasets and complex surface defect patterns.

In this study, YOLO12 and YOLO13 architectures are investigated for the detection and localization of surface defects on hot-rolled steel plates using the publicly available NEU-DET dataset, which contains 1,800 labeled images representing six defect categories. While the original dataset configuration allocated approximately 94.4% of the images for training and 5.6% for validation (16.7:1 training-to-validation ratio), this study also evaluated an alternative partition with 83.3% training and 16.7% validation (5:1 training-to-validation ratio). This alternative configuration was introduced to investigate the effect of validation set size on model performance and to provide a more reliable performance assessment by increasing the number of validation samples.

Moreover, the effect of input image resolution on defect detection performance was investigated. Since the original images in the dataset were provided at a resolution of 200×200 pixels, experiments were conducted using resized inputs of 224×224 and 640×640 pixels. In this

* **Corresponding Author:** ademdilbaz38@gmail.com

study, the lower-resolution setting was selected to preserve the characteristics of the original dataset while reducing computational requirements, whereas the standard YOLO input resolution was evaluated based on the conventional YOLO configuration, which generally provides improved capability for detecting small-scale defects. This comparative analysis aims to reveal the trade-off between detection accuracy and computational efficiency, providing practical insights into optimal preprocessing strategies for industrial surface defect inspection systems.

1.1. Literature Review

Wang et al. developed a deep learning-based framework for steel surface defect detection by integrating an improved ResNet50 classifier with an enhanced Faster R-CNN detector. The proposed method achieved an accuracy of 98.2% on a factory-based dataset and highlighted strong capability in defect classification and localization. The results highlighted the effectiveness of deep learning approaches for automated steel surface inspection and quality control applications [1].

Konovalenko et al. used steel surface defect classification using a deep residual neural network (ResNet)-based approach. The proposed method effectively extracted discriminative features from steel surface images and achieved an overall classification accuracy of 96.91% on the test dataset. The results indicated the potential of deep residual networks for automated defect recognition; however, the study mainly focused on defect classification rather than defect localization [2].

Kripalani investigated casting defect detection using deep learning-based approaches optimized with the PaddlePaddle YOLOv3 algorithm and CNN models. The study employed an augmented dataset obtained from NiTinol-based sensor measurements and evaluated defect detection performance using comparative analysis. The CNN model achieved an F1-score of 0.9939, while the improved R-CNN approach with spatial pyramid pooling reached an accuracy of 98.76%. Besides, the proposed methodology reduced false positive and false negative detections, demonstrating the potential of YOLO-based object detection frameworks for automated non-destructive testing and industrial defect inspection applications [3].

Lin et al. proposed an improved YOLO12-based framework for steel surface defect detection. The study introduced a lightweight multi-scale feature extraction module, an attention-guided feature fusion mechanism, and an optimized loss function to improve the detection of small and complex defects. Experimental results on the GC10-DET dataset showed that the baseline YOLO12n achieved a precision (P) of 63.5%, mAP@50 of 63.3%, and mAP@50–95 of 26.8%, while the integration of the proposed modules improved the mAP@50–95 value to 32.1%. The results demonstrated the effectiveness of the proposed improvements in enhancing feature representation and bounding box regression, highlighting the potential of recent YOLO architectures for steel surface defect inspection [4].

Fei et al. proposed an enhanced YOLO11-based framework (EDS-YOLO) for steel surface defect detection. The proposed method introduced improved feature fusion and lightweight convolution modules to enhance multi-scale defect representation and reduce the impact of low-contrast and complex background conditions. Experimental results indicated that EDS-YOLO achieved a precision of 89.8%, a recall (R) of 83.7%, a mAP@50 of 92.1%, and an mAP@50–95 of 68.7%, outperforming the baseline YOLO11 model. The results confirmed the effectiveness of improved YOLO architectures for detecting complex steel surface defects [5].

Liu proposed an improved YOLOX-based framework for steel surface defect detection. The proposed method incorporated a CSP CrossLayer module to enhance feature extraction, a Shuffle Attention mechanism to improve feature representation, and a PSBlock module to achieve more efficient multi-scale feature fusion. Experimental results indicated that the proposed model achieved a mAP@50 of 77.0% on the NEU-DET dataset and 88.8% on the steel rail defect dataset, with detection speeds of 100 FPS and 93 FPS, respectively. The study confirmed the effectiveness of improved YOLO-based detectors for accurate and efficient steel surface defect inspection [6].

Liu et al. proposed YOLO13-EnhCrack, an enhanced YOLO13-based framework for detecting low-contrast steel surface cracks. The proposed method incorporated an edge enhancement module, a dual-branch feature fusion strategy, and an optimized localization mechanism to improve crack representation under challenging industrial conditions. Experimental results demonstrated that the proposed model achieved the mAP@50 of 94.7% and the F1-score of 91.3%, outperforming the baseline YOLO13 model while providing more accurate localization of low-contrast cracks. The study confirmed the effectiveness of enhanced YOLO13 architectures for robust steel surface crack detection [7].

Dilbaz and Ozkan investigated simulation-based spot welding inspection on automotive chassis using YOLO-powered image processing. YOLOv3-s and YOLOv5-m architectures were evaluated for the detection of welding-related defects in a virtual inspection environment. Experimental results showed that the YOLOv5-m model outperformed YOLOv3-s, achieving an approximately 8–9% higher confidence score (CS ≈ 0.58) and improving the F1-score for critical explosion weld defects by approximately 5–6%, reaching 0.85. The findings revealed the applicability of YOLO-based object detection models for accurate localization and classification of welding

defects in automated quality inspection systems [8].

Ren et al. used DDR-YOLO, an enhanced steel surface defect detection framework based on YOLO11s. The proposed model incorporated a dual dynamic residual (DDR) module to improve feature extraction and multi-scale defect representation while maintaining computational efficiency. The study reported findings showing that the proposed model improved the mAP@50 by 3.6% on the NEU-DET dataset and 3.9% on the GC10-DET dataset compared with the baseline YOLO11s model, achieving higher detection accuracy without sacrificing inference speed. These results confirmed the effectiveness of lightweight YOLO11-based architectures for industrial steel surface defect inspection [9].

Lu and Qu used an improved YOLOv8-based framework (YOLO-SSD) for steel surface defect detection. The proposed model integrated SPD-Conv, the CBAM attention mechanism, and a lightweight C2f-Ghost module to enhance feature extraction while reducing computational complexity. The evaluation results showed that the proposed method achieved an mAP@50 of 79.8%, representing an improvement of 2.7% over the baseline YOLOv8 model, while maintaining efficient detection speed. The study demonstrated the effectiveness of lightweight attention-enhanced YOLO architectures for automated steel surface defect inspection [10].

Chen and Tang proposed an improved lightweight YOLO-based framework for steel surface defect detection. The proposed model enhanced feature extraction and multi-scale feature fusion to improve the detection of small surface defects while reducing computational complexity. The evaluation results showed that the proposed method achieved an mAP@50 of 95.4% and improved detection accuracy by approximately 2.8% compared with the baseline model, while maintaining real-time inference performance. The study revealed the effectiveness of lightweight YOLO architectures for automated steel surface inspection [11].

Wu and Zhang proposed SDS-YOLO11s, a lightweight steel surface defect detection framework based on YOLO11s. The proposed model incorporated an enhanced feature interaction module and a content-aware upsampling strategy to improve multi-scale feature representation while reducing computational complexity. The proposed method achieved improvements of 3.8% in mAP@50 on the NEU-DET dataset and 3.4% on the GC10-DET dataset compared with the baseline YOLO11s model, while maintaining efficient inference performance. The findings demonstrated the effectiveness of lightweight YOLO11-based architectures for accurate and efficient steel surface defect inspection [12].

Qin et al. proposed a Hierarchical Depth-Aware YOLO framework for efficient metal surface defect detection. The proposed model introduced a hierarchical depth-aware feature refinement strategy to enhance multi-scale feature representation while maintaining computational efficiency. The evaluation results showed that the proposed method achieved the mAP@50–95 of 72.6%, representing an improvement of 3.8% over the baseline YOLO model while reducing computational overhead. The study highlighted that depth-aware feature refinement can effectively improve the detection of complex metal surface defects under challenging industrial conditions [13].

Tang et al. used CPFS-YOLO, a cross-stage and multi-scale feature fusion framework for steel surface defect detection. The proposed model integrated cross-stage feature fusion and multi-scale feature aggregation modules to enhance feature representation and improve the detection of small and low-contrast defects. The evaluation results showed that the proposed method increased the mAP@50–95 from 26.8% to 32.1% compared with the baseline YOLO12n model, while also improving precision and recall. The study demonstrated the effectiveness of multi-scale feature fusion for enhancing steel surface defect detection performance [14].

Wang et al. proposed an enhanced YOLOv5-based framework for steel surface defect detection to improve the identification of small-scale defects. The proposed method integrated a Res2Block module into the backbone and a recursive gated convolution structure into the neck to enhance feature extraction and computational efficiency. The evaluation results showed that the proposed model achieved a mAP@50 of 67.8% and an F1-score of 86.0%, outperforming the original YOLOv5 and other object detection methods. The study indicated the effectiveness of improved YOLOv5 architectures for accurate steel surface defect identification [15].

Literature review indicates that although recent studies have achieved promising results in steel surface defect detection using advanced YOLO-based architectures, most existing works focus on improving network structures through customized modules, attention mechanisms, or feature fusion strategies. However, the effects of dataset partitioning strategies, particularly validation set proportions, and input image resolution variations on model learning performance have received limited attention. Furthermore, comprehensive comparative analyses of the latest YOLO architectures under different validation configurations and image resolutions remain insufficient.

The main contributions of this study can be summarized as follows:

A comprehensive comparative evaluation of the Ultralytics-developed YOLO12 and the iMoonLab-proposed YOLO13 architectures is performed for steel surface defect detection using the same dataset and experimental conditions.

The impact of different validation set ratios on model performance is systematically investigated to analyze their

influence on evaluation reliability and learning behavior.

The effect of input image resolution variations (224×224 and 640×640) on detection accuracy and computational efficiency is analyzed comparatively.

The study provides practical insights into optimal preprocessing and validation strategies for improving the reliability of YOLO-based steel surface defect inspection systems.

The remainder of this paper is organized as follows. Section 2 describes the materials and methods, including the dataset, model architectures, and evaluation metrics. Section 3 presents the experimental results, followed by a discussion of the findings in Section 4. Finally, the main conclusions and future work are presented in Section 5.

2. MATERIAL AND METHODS

2.1. NEU-DET Dataset (NDD)

The NEU-DET dataset, an open-access benchmark dataset for steel surface defect detection, was used in this study. The dataset contains a total of 1,800 annotated images belonging to six different defect classes, with 300 images allocated to each class. In the original NDD configuration, approximately 94.4% of the 200 × 200 pixel images were allocated for training and 5.6% for validation, corresponding to a training-to-validation ratio of approximately 16.7:1 [16]. Additionally, in this study, the dataset partition was rearranged to achieve a more balanced evaluation setting, where 83.3% of the images were used for training and 16.7% for validation, corresponding to a training-to-validation ratio of approximately 5:1. As a result, the number of annotated defect instances in the validation set increased from 232 to 701. Representative examples of the six typical defect classes that can appear on steel surfaces are illustrated in Figure 1. These include crazing (crack-like defects), inclusion (foreign particle contamination), patches (irregular surface patches), pitted surface (localized corrosion pits), rolled-in scale (oxide layers formed during rolling), and scratches (mechanical surface damage).

Table 1 summarizes the distribution of the NDD dataset across the training and validation sets. The numbers of images, instances, and samples for each defect category are reported to provide a detailed overview of the dataset composition. The same dataset partition was consistently used for the first and second evaluations, whereas in the third evaluation, 200 images were reallocated from the training set to the validation set with an equal distribution across all defect categories. Further details regarding all evaluation protocols are provided in Section 3 (Experimental Setup).

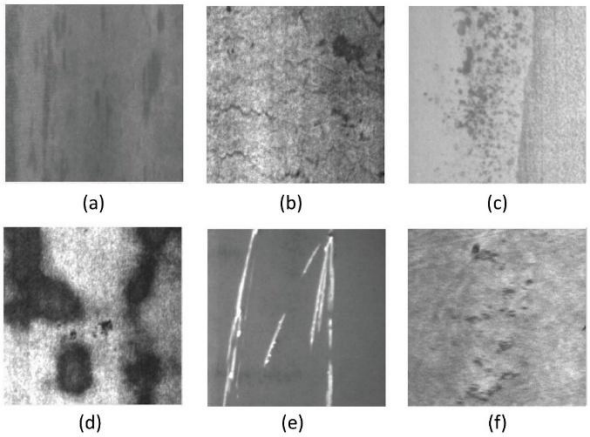

Figure 1. Sample Images of NDD (a) Inclusions, (b) Crazing, (c) Pitted surface, (d) Patches, (e) Scratches, and (f) Rolled in Scale.

Table 1. Distribution of the Instances in the NDD for the Six Defect Classes

Set	Images	Instances	Crazing	Inclusion	Patches	Pitted Surface	Rolled-in Scale	Scratches
Training	1,700	3,957	651	955	840	403	587	521
Validation	100	232	38	56	41	29	41	27
Total	1,800	4,189	689	1,011	881	432	628	548

The images in the NEU-DET dataset are stored in .jpg format and represented in the RGB color space. Each image contains one or more steel surface defects annotated using bounding boxes. The annotated defects exhibit considerable variations in shape, size, appearance, and spatial distribution, making the dataset challenging for object detection tasks. These characteristics have established NEU-DET as a widely adopted benchmark for the development, evaluation, and comparative analysis of machine learning and deep learning-based object detection methods for automated steel surface defect detection.

2.2. YOLO12 Architecture

YOLO12 is a state-of-the-art one-stage object detector based on an attention-centric architecture proposed by Tian et al. and officially implemented in the Ultralytics framework [17]. By integrating efficient feature extraction, multi-scale feature representation, and an optimized detection head within a unified framework, YOLO12 achieves high detection accuracy while maintaining low computational complexity and fast inference speed. Owing to these characteristics, it is well suited for industrial visual inspection applications, including automated steel surface defect detection. The architecture consists of three main components: the backbone, which extracts hierarchical feature representations from the input image; the neck, which aggregates multi-scale features to improve the detection of objects with varying sizes; and the detection head, which

predicts object classes, confidence scores, and bounding box coordinates [18].

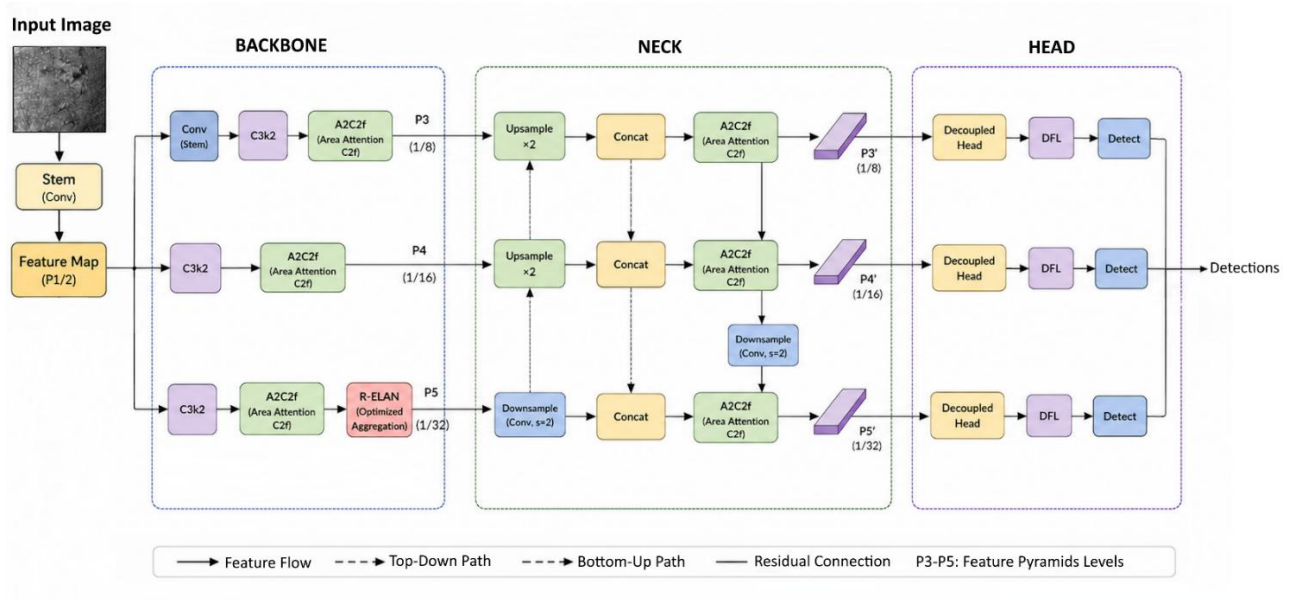

Figure 2. Layers and Implementation Stages of the YOLO12 Architecture

Figure 2 illustrates the overall architecture of the YOLO12 model. The input image is first processed by the backbone to extract hierarchical features, followed by the neck, where multi-scale feature fusion is performed using FPN and PAN structures. Finally, the detection head predicts the object classes and corresponding bounding boxes for steel surface defects.

The backbone of YOLO12 begins with convolution (Conv) layers for low-level feature extraction, followed by C3k2 blocks that capture hierarchical image features. A key improvement is the replacement of conventional C2f modules with Area Attention C2f (A2C2f) blocks, which integrate a FlashAttention-based area attention mechanism to enhance feature representation [19]. Finally, an R-ELAN module is employed to simplify feature aggregation, improve optimization efficiency, and accelerate inference.

The neck of YOLO12 employs C3k2 blocks to refine multi-scale feature representations while improving channel-wise feature processing. Upsampling layers increase the spatial resolution of deep semantic features, enabling effective fusion with high-resolution shallow features. Feature maps are merged through concatenation (Concat) operations following the PAN-FPN architecture, while convolution (Conv) layers further refine the fused features and synchronize channel dimensions before detection.

The head of YOLO12 adopts a decoupled head architecture that independently performs classification and bounding box regression, improving optimization efficiency [20]. Distribution Focal Loss (DFL) is employed to achieve more accurate bounding box localization, while the final detection layer predicts object classes, confidence scores, and bounding box coordinates. The model uses the SiLU activation function throughout the network to enhance feature learning, and Complete Intersection over Union (CIoU) loss is utilized for robust bounding box regression during training.

2.3. YOLO13 Architecture

YOLO13, publicly available through iMoonLab, is an advanced one-stage object detection architecture designed to improve feature representation and detection performance while maintaining computational efficiency [21]. The model introduces enhanced attention-based feature extraction and optimized feature fusion to better capture spatial information across multiple scales. Similar to all previous YOLO architectures, YOLO13 consists of three main components: the backbone for hierarchical feature map extraction, the neck for multi-scale feature aggregation, and the detection head for object classification and bounding box prediction.

Unlike YOLO12, which employs A2C2f and R-ELAN modules for attention-enhanced feature extraction, YOLO13 integrates the HyperACE module to model high-order spatial correlations through hypergraph-based feature learning, enabling richer feature representations. YOLO13 replaces the conventional PAN-FPN feature fusion strategy of YOLO12 with the FullPAD architecture, which provides fine-grained feature aggregation and distribution through multiple information flow pathways, improving multi-scale feature integration.

The detection head of YOLO13 retains the decoupled head and DFL framework while incorporating FullPAD-enhanced features and an optimized one-to-many assignment strategy to improve localization accuracy and classification performance [22]. Table 2 summarizes the

key architectural differences between YOLO12 and YOLO13. To quantitatively evaluate the performance of these architectures, several standard object detection metrics were employed. In addition, confusion matrices were analyzed to provide a detailed assessment of class-wise prediction performance.

Table 2. Comparison of the Architectural Characteristics of YOLO12 and YOLO13

Feature	YOLO12	YOLO13
Core Design	Attention-centric architecture	Hypergraph- and Full-Pipeline-based architecture
Backbone	A2C2f with R-ELAN modules	HyperACE for hypergraph-based feature modeling
Neck	PAN-FPN with A2C2f-based feature fusion	FullPAD with three dedicated feature propagation pathways
Feature Representation	FlashAttention-based local and global feature enhancement	Hypergraph-based high-order spatial correlation modeling
Small and Large Object Detection	Balanced local–global feature learning	Enhanced multi-scale representation through high-order correlation modeling

2.4. Confusion Matrix

A confusion matrix was employed to evaluate the class-wise performance of the proposed models by comparing the predicted labels with the corresponding ground-truth annotations [23]. The confusion matrices provide a direct visualization of correct and incorrect predictions for each defect category, enabling the identification of well-classified classes as well as defect categories that remain challenging for the models. The resulting confusion matrices are presented in the Experimental Results section to support the qualitative analysis of class-wise detection performance.

2.5. Performance Metrics

The performance of the object detection models was evaluated using the confusion matrix, which compares predicted labels with their corresponding ground-truth labels. The confusion matrix provides the numbers of true positives (TP), false positives (FP), false negatives (FN), and true negatives (TN), forming the basis for calculating evaluation metrics such as precision, recall, and mAP metrics. These metrics provide a comprehensive assessment of the classification and localization performance of the models.

Table 3. Performance Metrics, Formulas, and Explanation

Performance Metrics	Math Formulas	Explanation
Precision (P)	$\frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FP}}$	The proportion of predicted positive detections that are actually correct
Recall (R)	$\frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FN}}$	The proportion of actual positive instances that are correctly detected
AP	$\int_0^1 \mathrm{P(R)}\,\mathrm{dR}$	The area under the PR curve for a single object class
mAP	$\frac{1}{\mathrm{N}_C}\sum_{i=1}^{\mathrm{N}_C}\mathrm{AP}_i$	The average precision calculated across all object classes

The performance of both YOLO models was evaluated using precision, recall, AP, and mAP. The mathematical formulations and corresponding descriptions of these evaluation metrics are presented in Table 3. The mAP@50–95 metric was calculated by averaging the AP values over IoU thresholds ranging from 0.50 to 0.95 with a step size of 0.05, following the COCO evaluation protocol.

2.6. YOLO Variants

YOLO models are available in different scales, including Nano (N), Small (S), Medium (M), Large (L), and Extra Large (X), which differ in network depth, width, parameter count, and computational complexity while maintaining the same architecture [24]. Lightweight models (Nano and Small) require fewer parameters and generally provide better generalization on limited datasets, whereas Large and XLarge models are more suitable for large-scale datasets due to their higher representational capacity

Although larger YOLO models provide higher feature extraction capacity, the increased number of parameters does not always lead to improved performance on limited-scale datasets. A similar observation was reported by Liu and Juang in their comparative analysis of YOLOv8 variants, where the largest model (YOLOv8-X) did not outperform YOLOv8-L despite having higher model capacity. This finding indicates that increasing model complexity does not necessarily guarantee accuracy improvement and highlights the importance of balancing model capacity with the available training data [25].

3. EXPERIMENTAL RESULTS

All experiments were performed in the Google Colab environment. The experimental configurations adopted in this study are presented in Table 4. All models were optimized using the AdamW optimizer with an initial learning rate of 0.001 and a momentum value of 0.937. The weight decay was applied through parameter groups, including bias and normalization-related parameters with zero decay and weight parameters with a decay factor of 0.0005. The batch size was fixed at 8. Each model was

trained for 50 epochs using the same optimization strategy to provide a consistent and unbiased comparison among the evaluated models.

To maintain a fair and consistent comparison, all baseline models were initialized with the official pretrained weights provided by the Ultralytics framework (v8.4.98). The YOLO12 and YOLO13 models were initialized using their respective pretrained checkpoints, while preserving their original network architectures. All baseline models were trained and evaluated under the same experimental conditions without any additional architectural modifications.

Table 4. Experimental setup configuration for all models

Component	Specification
Operating System	Google Colab (Linux)
GPU	NVIDIA Tesla T4 (14 GB VRAM)
Python	3.12.13
PyTorch	2.10.0 + CUDA 12.8 (torch-2.10.0+cu128)
CUDA	CUDA 12.8
Ultralytics	V8.4.98

The comparative analysis of four models in terms of parameters and FLOPs is presented in Table 5. The evaluated models include YOLO12 and YOLO13 variants with different model scales (small and large). All models were initialized with official pretrained weights and trained on the same domain-specific dataset using identical hyperparameter settings. The obtained architectural characteristics highlight the differences in model complexity and computational requirements among the evaluated YOLO variants. The comparative evaluation results based on standard object detection metrics are presented in the following section.

Table 5. Architectural Characteristics of the Four YOLO-Based Models

Model	Pre-trained	Layers	Params (M)	Flops (G)
YOLO12-S	yolov12s	272	9.3	21.4
YOLO12-L	yolov12l	488	26.4	88.9
YOLO13-S	yolo13s	203	9.0	20.8
YOLO13-L	yolo13l	379	27.6	88.4

After being trained under the same experimental conditions, all models were evaluated using commonly adopted object detection metrics. The quantitative comparisons, performance analysis, and qualitative evaluation results are discussed in the following section.

The YOLO12 and YOLO13 models were trained and evaluated under identical experimental settings with all evaluations conducted using random seeds 42, 43 and 44. For the first evaluation, an input image size of 640 × 640 pixels was used. The dataset was divided into training and validation sets with a 16.7:1 ratio (94.4% training and 5.6% validation). Model performance was quantitatively assessed using standard object detection metrics, while qualitative comparisons were conducted using representative validation images. The confusion matrices obtained for the YOLO12-S, YOLO12-L, YOLO13-S, and YOLO13-L are presented in Figure 3.

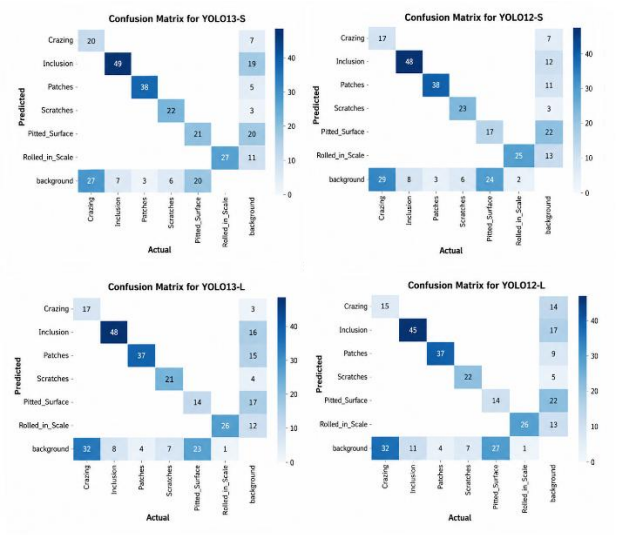

Figure 3. Comparative Confusion Matrices of the YOLO-Based Models Obtained in the First Evaluation Using Seed 42

Among the evaluated models, YOLO13-S achieved the highest numbers of true positive predictions for the Crazing, Pitted Surface, and Rolled-in Scale classes, whereas YOLO12-S obtained the highest TP prediction for the Scratches class.

In the second evaluation, the four YOLO models were trained and evaluated under identical experimental settings using an input image size of 224 × 224 pixels. The dataset was divided into training and validation sets with a ratio of 16.7:1. The confusion matrices obtained for the four YOLO models are presented in Figure 4.

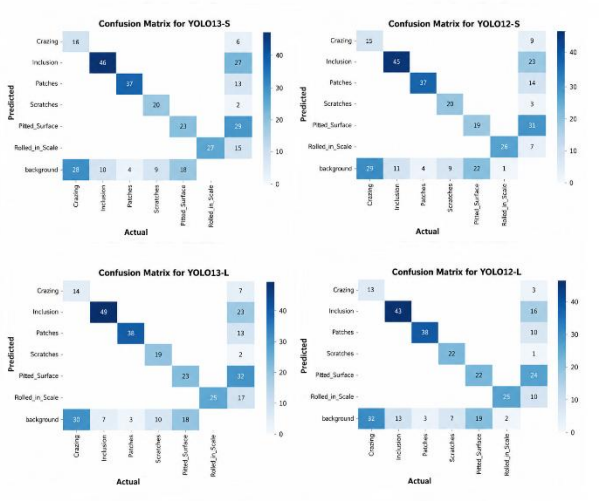

Figure 4. Comparative Confusion Matrices of the YOLO-Based Models Obtained in the Second Evaluation Using Seed 42

Among the evaluated models, YOLO13-S obtained the highest TP count for the Crazing and Rolled-in Scale classes. To further investigate the effect of the input configuration on model performance, an additional evaluation was conducted by modifying the input image size and the training-to-validation ratio.

In the third evaluation, the same four YOLO models were trained and evaluated under identical experimental settings using an input image size of 224 × 224 pixels. The dataset was divided into training and validation sets with a 5:1 ratio. The confusion matrices obtained for these models are presented in Figure 5.

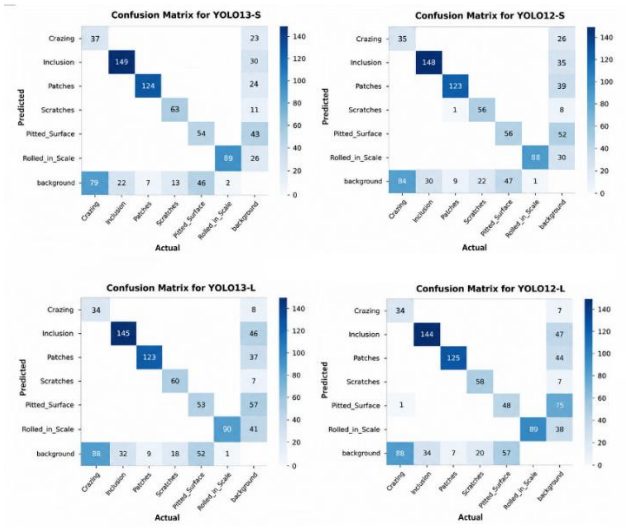

Figure 5. Comparative Confusion Matrices of the YOLO-Based Models Obtained in the Third Evaluation Using Seed 42

YOLO13-S achieved the highest TP predictions for the Crazing, Inclusion, and Scratches classes. In addition, the confusion matrices showed only one wrong class prediction in each of the YOLO12-S and YOLO12-L models, indicating a very limited number of class misclassifications. Overall, the confusion matrices demonstrate that all evaluated models exhibit strong class discrimination capability, with only minor differences across defect categories.

Moreover, Figure 6 illustrates the evolution of the training classification loss for the evaluated YOLO12 and YOLO13 models over 50 training epochs. All models exhibited stable and consistent convergence behavior, with a gradual reduction in classification loss throughout the training process. YOLO13-S achieved the lowest final classification loss (0.967), indicating the most effective optimization of class-specific features during training. This was followed by YOLO12-S with a final loss of 1.038, while YOLO13-L and YOLO12-L converged to final classification loss values of 1.179 and 1.230, respectively. These results indicate that YOLO13-S exhibited more favorable convergence behavior than the other evaluated models under the same training configuration.

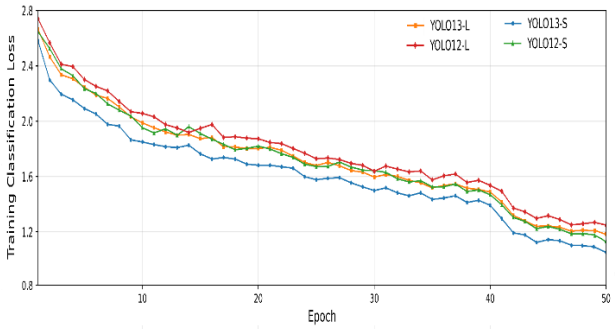

Figure 6. Comparative Classification Loss of Four YOLO-Based Models Using Seed 42

In addition to the confusion matrix analysis, comparative P–R curve analyses were performed for the four YOLO-based model variants under three different experimental settings, considering different input image sizes and training-to-validation ratios. All P–R curves presented in this section correspond to the experiments conducted using random seed 42. These curves provide a comprehensive evaluation of the trade-off between precision and recall across different confidence thresholds. The results of the first evaluation, conducted using an input image size of 640 × 640 pixels and a training-to-validation ratio of 16.7:1, are presented in Figure 7.

YOLO13-S achieved the highest overall performance with a mAP@50 of 0.753 using seed 42, followed by YOLO12-S (0.746, seed 42), YOLO13-L (0.74, seed 43), and YOLO12-L (0.734, seed 44)The P–R curves indicate that all models achieved high detection performance for the Inclusion, Patches, Scratches, and Rolled-in Scale classes, whereas comparatively lower AP values were observed for the Crazing and Pitted Surface classes.

In contrast, the results of the second evaluation, conducted using an input image size of 224 × 224 pixels and a training-to-validation ratio of 16.7:1, are presented in Figure 8.

The results of the third evaluation, conducted using an input image size of 224 × 224 pixels but with a training-to-validation ratio of 5:1, are presented in Figure 9.

Compared with the first evaluation, all models exhibited lower mAP@50 values, while the performance differences among the four model variants became considerably smaller. Nevertheless, all models maintained consistent P-R characteristics across the evaluated defect classes. Among the evaluated models, YOLO13-S again achieved the highest overall performance, with a mAP@50 of 0.728.

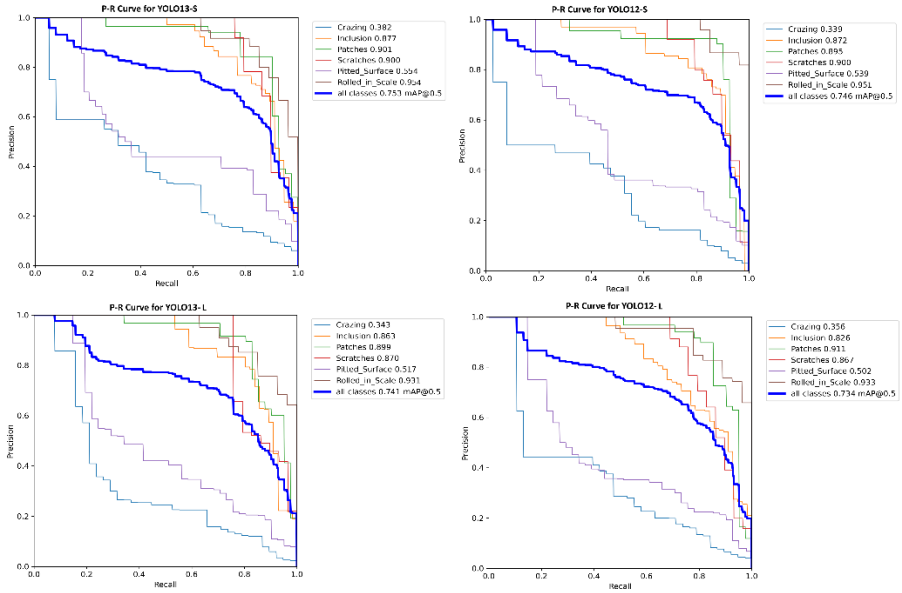

Figure 7. Comparative P–R Curves of the YOLO-Based Models Obtained in the First Evaluation Using Seed 42

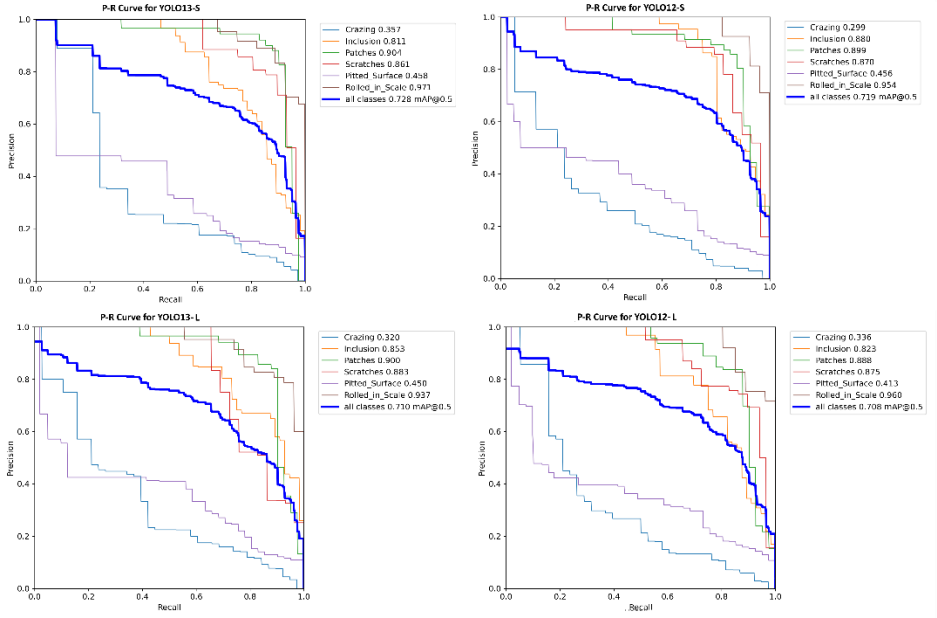

Figure 8. Comparative P–R Curves of the YOLO-Based Models Obtained in the Second Evaluation Using Seed 42

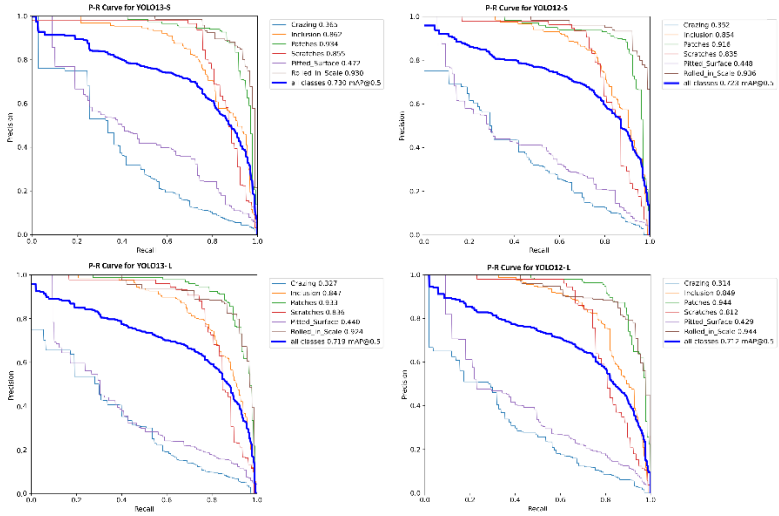

Figure 9. Comparative P–R Curves of the YOLO-Based Models Obtained in the Third Evaluation Using Seed 42

Compared with the second evaluation, all models demonstrated improved mAP@50 performance, although the performance differences among the four model variants remained relatively small. However, the overall mAP@50 values were still lower than those obtained in the first evaluation. Additionally, all models maintained consistent P–R characteristics across the evaluated defect classes. Among the evaluated models, YOLO13-S again achieved the highest overall performance, with a mAP@50 of 0.730.

First, the class-wise analysis, Table 6 presents the validation performance of the YOLO13-S model for the first evaluation (Seed = 42). The table reports the Precision, Recall, AP@50, and AP@50–95 values for each defect category, highlighting the detection performance across different surface defect classes. The corresponding results were obtained using a 1700:100 training/validation split with an input image size of 640×640 pixels.

Table 6. Class-wise validation performance metrics of the YOLO13-S model for the first evaluation

Class	Images	Instances	Precision	Recall	AP@50	AP@50–95
Crazing	16	41	0.640	0.316	0.462	0.226
Inclusion	18	48	0.747	0.839	0.878	0.509
Patches	16	46	0.736	0.886	0.887	**0.596**
Rolled-in Scale	17	29	**0.838**	0.759	0.892	0.569
Pitted Surface	16	41	0.539	0.475	0.496	0.272
Scratches	17	27	0.712	**0.916**	**0.907**	0.563

Among the six defect classes, Rolled-in Scale achieved the highest precision (0.838), whereas Scratches obtained the highest recall (0.916) and AP@50 (0.907). The highest mAP@50–95 was achieved by the Patches class (0.596). In contrast, Crazing exhibited the lowest recall (0.316), AP@50 (0.462), and AP@50–95 (0.226), indicating that it remained the most challenging defect category to detect.

To provide a more detailed second class-wise analysis, Table 7 presents the validation performance of the YOLO13-S model (Seed = 42). The table reports the Precision, Recall, AP@50, and AP@50–95 values for each defect category, highlighting the detection performance across different surface defect classes. The corresponding results were obtained using a 1700:100 training/validation split with an input image size of 224×224 pixels.

Table 7. Class-wise validation performance metrics of the YOLO13-S model for the second evaluation

Class	Images	Instances	Precision	Recall	AP@50	AP@50–95
Crazing	16	41	0.511	0.236	0.371	0.173
Inclusion	18	48	0.749	0.804	0.871	0.509
Patches	16	46	0.782	0.902	**0.897**	**0.590**
Rolled-in Scale	17	29	**0.858**	0.771	0.895	0.498
Pitted Surface	16	41	0.489	0.435	0.418	0.201
Scratches	17	27	0.709	**0.908**	0.893	0.553

For the six defect classes, Rolled-in Scale achieved the highest precision (0.858), while Scratches obtained the highest recall (0.908). Patches achieved the highest AP@50 (0.897) and AP@50–95 (0.590). Crazing again showed the most challenging detection performance, exhibiting the lowest recall, AP@50, and AP@50–95 values, although its precision was not the lowest among the six classes.

The third class-wise analysis, Table 8 presents the validation performance of the YOLO13-S model (Seed = 42). The table reports the Precision, Recall, AP@50, and AP@50–95 values for each defect category, highlighting the detection performance across different surface defect classes. The corresponding results were obtained using a 1500:300 training/validation split with an input image size of 224×224 pixels.

Table 8. Class-wise validation performance metrics of the YOLO13-S model for the third evaluation

Defect Class	Images	Instances	Precision	Recall	AP@50	AP@50–95
Crazing	48	120	0.608	0.243	0.385	0.189
Inclusion	54	158	0.742	0.809	0.847	0.470
Patches	50	139	0.719	**0.909**	**0.913**	**0.615**
Rolled-in Scale	48	82	**0.885**	0.744	0.846	0.513
Pitted Surface	49	112	0.502	0.460	0.447	0.234
Scratches	51	90	0.718	0.901	0.912	0.548

In the final evaluation, Rolled-in Scale achieved the highest precision (0.885), while Patches obtained the highest recall (0.909), AP@50 (0.913), and AP@50–95 (0.615). Crazing again showed the most challenging detection performance, exhibiting the lowest recall, AP@50, and AP@50–95 values, except for precision.

Table 9. Comparative P, R, mAP@50, and mAP@50–95 Values of the YOLO-Based Models Across (Mean ± Std Over Three Runs)

Model	Evaluation	Precision	Recall	mAP@50	mAP@50-95
YOLO12-S	1st	0.802±0.026	0.714±0.019	0.738±0.010	0.428±0.013
YOLO12-S	2nd	0.818±0.023	0.675±0.024	0.709±0.014	0.401±0.019
YOLO12-S	3rd	0.784±0.018	0.701±0.025	0.720±0.013	0.419±0.022
YOLO12-L	1st	0.779±0.023	0.718±0.027	0.723±0.017	0.412±0.024
YOLO12-L	2nd	0.797±0.016	0.680±0.028	0.699±0.013	0.398±0.021
YOLO12-L	3rd	0.766±0.014	**0.740±0.018**	0.703±0.018	0.405±0.030
YOLO13-S	1st	0.811±0.032	0.718±0.024	**0.743±0.012**	**0.435±0.016**
YOLO13-S	2nd	**0.835±0.015**	0.686±0.022	0.720±0.016	0.409±0.021
YOLO13-S	3rd	0.791±0.025	0.709±0.024	0.728±0.018	0.421±0.019
YOLO13-L	1st	0.804±0.027	0.697±0.025	0.739±0.014	0.431±0.018
YOLO13-L	2nd	0.800±0.031	0.684±0.023	0.706±0.013	0.402±0.029
YOLO13-L	3rd	0.773±0.023	0.722±0.034	0.712±0.015	0.414±0.023

As summarized in Table 9, the experimental results are reported as mean ± standard deviation over three independent runs using random seeds 42, 43, and 44. Overall, YOLO13-S consistently demonstrated the most balanced detection performance across the three evaluation settings. In the first evaluation, YOLO13-S achieved the highest mAP@50 score (0.753, seed 42) and the highest mAP@50–95 score (0.449, seed 43) among all evaluated models. Across all evaluations, the highest precision was obtained by YOLO13-S in the second evaluation (0.848, seed 44), whereas the highest recall was achieved by YOLO12-L in the third evaluation (0.752, seed 44).

For this experiment, the four YOLO-based models evaluated in this study were trained for 100 epochs using seed 42 under the first evaulation setting. The corresponding performance results are presented in Table 10.

Table 10. Comparative P, R, mAP@50, and mAP@50–95 Values of the YOLO-Based Models at Different Training Epochs

Model	Epoch	Precision	Recall	mAP@50	mAP@50-95
YOLO12-S	50	0.828	0.729	0.746	0.439
YOLO12-S	100	0.840	0.753	0.788	0.486
YOLO12-L	50	0.801	0.741	0.734	0.430
YOLO12-L	100	0.817	**0.775**	0.774	0.477
YOLO13-S	50	0.839	0.733	0.753	0.449
YOLO13-S	100	**0.848**	0.760	**0.800**	0.498
YOLO13-L	50	0.817	0.719	0.741	0.447
YOLO13-L	100	0.842	0.750	0.785	**0.502**

Early stopping was configured with a patience of 30 epochs; however, early stopping was not triggered during training, as the validation performance continued to improve within the specified patience interval. Therefore, all models completed the planned 100 epochs. At 100 epochs, YOLO12-L again achieved the highest recall (0.775) among all evaluated models, reaching this value at epoch 97 and representing an increase of 3.4 percentage points (pp) compared with 50 epochs. YOLO13-S again achieved the highest precision (0.848) and mAP@50 (0.800), reaching these values at epochs 79 and 90, respectively. These results correspond to increases of 0.9 pp and 4.7 pp, respectively, compared with 50 epochs. In terms of mAP@50–95, YOLO13-S achieved the highest value at 50 epochs (0.449); however, at 100 epochs, YOLO13-L surpassed YOLO13-S, achieving the highest mAP@50–95 value (0.502) at epoch 98, corresponding to an increase of 5.5 pp. These results indicate that the larger models, particularly the L variants, benefited from extended training and exhibited improved convergence with additional epochs.

Table 11. Training and Validation Loss Values of Larger YOLO Models Across Different Numbers of Epochs

Model	Epoch	Training Box Loss	Training Class Loss	Training DFL Loss	Validation Box Loss	Validation Class Loss	Validation DFL Loss
YOLO12-L	50	1.40087	1.32158	1.67231	1.46575	1.34787	1.72256
YOLO12-L	100	1.12333	0.97572	1.49730	1.39430	1.12006	1.66928
YOLO13-L	50	1.29220	1.26897	1.69833	1.42353	1.12419	1.74912
YOLO13-L	100	1.08321	0.92653	1.50922	1.36672	1.04290	1.68361

From 50 to 100 epochs, the training classification loss decreased by 26.2% for YOLO12-L and by 27.0% for YOLO13-L, as shown in Table 11. Thus, both models

exhibited a comparable reduction in classification loss, with YOLO13-L showing a slightly greater relative improvement. At 100 epochs, the training box loss of YOLO13-L decreased by 16.17%, while the training DFL loss decreased by 11.13%. Among these loss components, the training class loss exhibited the greatest reduction for both models.

In addition to the confusion matrix and P–R curve analyses, the detection performance of the best-performing model (YOLO13-S) was further evaluated using confidence scores and IoU-based bounding box predictions. The detection results obtained on the validation set are presented in Figure 10. The evaluated model achieved the shortest inference times ranging from 14.6 ms to 24.9 ms per image, demonstrating high computational efficiency for steel surface defect detection. In comparison, the YOLO13-L model exhibited inference times ranging from 22.6 ms to 58.1 ms per image, reflecting the increased computational cost associated with its larger network architecture. Similarly, the YOLO12-S achieved inference times ranging from 15.1 ms to 25.8 ms per image, whereas the YOLO12-L model required between 20.9 ms and 52.0 ms per image.

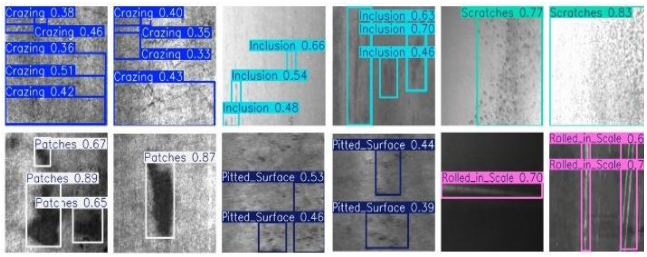

Figure 10. Representative Detection Results of YOLO13-S on the NDD Validation Set

Among the evaluated defect classes, the lowest confidence scores and localization performance were observed for the Crazing class. In contrast, the Scratches class achieved the highest confidence scores with highly accurate bounding box localization, indicating superior detection performance. These CS and IoU-based detection results are consistent with the quantitative findings obtained from the confusion matrix and P–R curve analyses, further confirming the robustness and reliability of the YOLO13-S model for steel surface defect detection.

4. RESULTS AND DISCUSSION

The proposed study comparatively evaluated four YOLO-based models (YOLO12-S, YOLO12-L, YOLO13-S, and YOLO13-L) under three different experimental settings by varying the input image size and the training-to-validation ratio. Across all evaluations, YOLO13-S consistently demonstrated the best overall detection performance, achieving the highest mAP@50 values of 0.743 ± 0.012 for three different runs. Furthermore, YOLO13-S obtained the highest mAP@50–95 value of 0.435 ± 0.016 across three independent runs. The confusion matrix, P–R curve, and confidence score analyses further confirmed the robustness and reliability of YOLO13-S for steel surface defect detection.

In the third evaluation, which included the largest validation set among all evaluations, the confusion matrix analysis of YOLO13-S yielded 516 TP, 157 FP, and 169 FN. Compared with YOLO13-S, YOLO12-S produced 506 TP (1.94% fewer), YOLO12-L achieved 498 TP (3.49% fewer), and YOLO13-L recorded 505 TP (2.13% fewer). The TP-based class-wise detection rates of YOLO13-S were as follows: Rolled-in Scale achieved the highest detection performance with 97.8%, followed by Patches (94.7%), Inclusion (87.1%), Scratches (82.9%), Pitted Surface (54.0%), and Crazing (31.9%). Furthermore, only one false positive in each of YOLO12-L and YOLO13-L resulted from inter-class misclassification, indicating that nearly all false positives were associated with the background class rather than confusion among defect categories. These results demonstrate that YOLO13-S achieved the highest overall TP count among the evaluated models, while the class-wise analysis revealed strong detection capability for Rolled-in Scale and Patches but lower detection performance for Crazing and Pitted Surface.

The experimental results demonstrate that both the YOLO12 and YOLO13 architectures provide reliable performance for steel surface defect detection under different experimental settings. Among the evaluated models, YOLO13-S consistently achieved the highest overall detection performance by obtaining the best mAP@50 and mAP@50–95 values while maintaining competitive precision and recall. This superior performance may be associated with the architectural characteristics of YOLO13. Unlike YOLO12, which employs A2C2f and R-ELAN modules for attention-based feature extraction, YOLO13 incorporates the HyperACE backbone and the FullPAD feature aggregation architecture. These architectural differences may have contributed to richer high-order spatial feature representation and more effective multi-scale information fusion, which may have improved the detection of complex defect patterns while preserving localization accuracy. However, the individual contributions of these components were not investigated in this study.

An interesting observation is that the smaller YOLO13-S model consistently outperformed the larger YOLO13-L model under all experimental settings. Although larger models generally possess higher representational capacity, this advantage does not necessarily translate into better performance on relatively small datasets such as NEU-DET. The increased number of parameters may lead to reduced generalization capability and a higher tendency to overfit the training data. In contrast, the lightweight YOLO13-S model appears to provide a more appropriate

balance between model capacity and dataset size, resulting in improved generalization and more robust detection performance. Similar observations have also been reported in previous comparative studies on different YOLO variants, where increasing model complexity did not always produce higher detection accuracy.

The experimental results further indicate that both the input image resolution and the training-to-validation ratio influence model performance. Using an input image size of 640 × 640 pixels generally improved detection accuracy for all evaluated models by preserving more discriminative spatial information, particularly for small and irregular surface defects. Conversely, reducing the input resolution to 224 × 224 pixels resulted in lower mAP values and reduced performance differences among the evaluated models. Additionally, increasing the validation set from the original 5.6% to 16.7% provided a more reliable and representative performance assessment by increasing the number of validation samples, although the larger validation partition slightly reduced the amount of training data available for model optimization.

The confusion matrix, Precision–Recall curves, and confidence score analyses consistently supported these findings. Among the six defect categories, Scratches and Rolled-in Scale achieved the highest detection performance, whereas Crazing remained the most challenging class because of its irregular appearance, low contrast, and large intra-class variability. These characteristics make feature representation and localization considerably more difficult than for the remaining defect categories.

Overall, the results indicate that YOLO13-S provides the most balanced trade-off between detection accuracy, computational complexity, and generalization capability for steel surface defect detection. Although the experiments were conducted exclusively on the NEU-DET dataset, future work will focus on validating the proposed evaluation on additional benchmark datasets, as well as investigating the real-time deployment of both YOLO13-S and the lightweight YOLO13-N model in industrial inspection systems to further evaluate inference speed, robustness, and practical applicability.

5. CONCLUSIONS

This study presented a comprehensive comparative evaluation of four YOLO-based models for steel surface defect detection under three different experimental settings by varying the input image size and the training-to-validation ratio. The experimental results demonstrated that both YOLO12 and YOLO13 architectures are capable of accurately detecting surface defects, while the YOLO13 series consistently exhibited superior overall performance.

Among the evaluated models, YOLO13-S achieved the most balanced detection performance, consistently achieving the highest mAP values while maintaining competitive precision and recall under all evaluated experimental configurations. The confusion matrix, Precision–Recall curves, and confidence score analyses further confirmed its robust detection capability and reliable class discrimination. In particular, the Crazing class remained the most challenging defect category, followed by the Pitted Surface class, whereas the Scratches and Rolled-in Scale classes consistently achieved the highest detection performance.

The results also demonstrated that both the input image size and the training-to-validation ratio significantly influence detection accuracy. Higher input resolutions generally improved the overall detection performance, highlighting the importance of selecting appropriate experimental configurations for steel surface defect inspection.

Overall, this study provides practical insights into the influence of model architecture, input resolution, and dataset partitioning on the reliability of steel surface defect detection, while demonstrating the effectiveness of YOLO13-S for automated industrial inspection. Future work will focus on validating the proposed approach through cross-dataset experiments using additional benchmark datasets, such as GC10-DET and Severstal Steel Defect Detection, to further assess the generalization capability of the models. Furthermore, Explainable Artificial Intelligence (XAI) techniques, such as Grad-CAM and Eigen-CAM, will be incorporated to provide visual interpretations of the detection process, thereby improving model transparency and supporting more reliable industrial decision-making.

Declaration of Ethical Standards

The author declares that this study did not involve human participants or animals. All ethical principles related to authorship, citation, data use, and research integrity were followed.

CRediT Authorship Contribution Statement

Adem Dilbaz: Conceptualization, Methodology, Software, Investigation, Data Curation, Formal Analysis, Visualization, Writing – Original Draft, Writing – Review & Editing.

Declaration of Competing Interest

The author declares that there are no known competing financial interests or personal relationships that could have influenced the work reported in this paper.

Funding / Acknowledgements

The author received no external funding for this research.

Data Availability Statement

The dataset used in this study is publicly available on

Kaggle: https://www.kaggle.com/datasets/sovitrath/neu-steel-surface-defect-detect-trainvalid-split

References

[1] S. Wang, X. Xia, L. Ye, and B. Yang, "Automatic detection and classification of steel surface defects using deep convolutional neural networks," Metals, vol. 11, no. 3, pp. 388, 2021, doi: 10.3390/met11030388.

[2] I. Konovalenko, P. Maruschak, J. Brezinová, J. Viňáš, and J. Brezina, "Steel surface defect classification using deep residual neural network," Metals, vol. 10, pp. 846, 2020, doi: 10.3390/met10070846.

[3] K. Kripalani, "Experimental work on cast defect detection by nanoindentation machine using NiTinol wire sensors using augmented technique optimized by PP YOLOv3 algorithm comparative analysis procedure," Intelligent Methods in Engineering Sciences, vol. 2, no. 3, pp. 75–83, 2023.

[4] Y. Lin, Y. Zhang, D. Yang, W. Guan, D. Hou, Q. Wang, and W. Xing, "Steel surface defect detection based on YOLOv12," Engineering Letters, vol. 33, no. 12, pp. 4806–4817, 2025.

[5] Z. Fei, L. Liu, R. Zhao, C. Yang, Y. Li, and X. Xiang, "EDS-YOLO: An enhanced steel surface defect detection method based on improved YOLO11," Measurement Science and Technology, vol. 37, no. 13, p. 136204, 2026, doi: 10.1088/1361-6501/ae5271.

[6] T. Liu, "Enhanced Zero-Shot YOLOv10 for multi-class tiny-object detection of steel surface defects," in 2024 6th International Conference on Robotics and Computer Vision (ICRCV), 2024, pp. 44–52.

[7] W. Liu, S. Chen, S. Wang, W. Zhong, J. Wang, and H. Liao, "YOLOv13-EnhCrack for low-contrast steel surface crack detection," in International Conference on Machine Learning and Artificial Intelligence Applications (MLAIA 2025), SPIE, vol. 14134, Art. no. 1413402, 2026.

[8] I.A. Ozkan and A. Dilbaz, "Simulation-based spot welding inspection on automotive chassis using YOLO-powered image processing," International Journal of Automotive Engineering and Technologies, vol. 14, no. 3, pp. 170–180, 2025, doi: 10.18245/ijaet.1729908.

[9] X. Ren, Z. Liu, H. Cheng, and L. Chen, "DDR-YOLO: An efficient detection model based on YOLO11s for steel surface defect," IEEE Access, vol. 13, pp. 147202–147217, 2025, doi: 10.1109/ACCESS.2025.3599978.

[10] X. Y. Lu and M. X. Qu, "Steel surface defect detection based on improved YOLOv8," Engineering Research Express, vol. 7, no. 1, 2025, doi: 10.1088/2631-8695/ad9b9d.

[11] H. Chen and J. Tang, "Performance comparison and analysis of YOLOv8/v9/v10 in steel plate surface defect detection," Science and Technology of Engineering, Chemistry and Environmental Protection, vol. 1, no. 1, 2025, doi: 10.61173/6bnb5p03.

[12] F. Wu and Y. Zhang, "A lightweight YOLOv11-based model for steel surface defect detection," Engineering Letters, vol. 34, no. 7, p. 2728, 2026.

[13] Q. Qin, A. S. M. Khairuddin, N. Idros, H. Liu, L. Fan, Z. Yang, J. Li, and C. Zhu, "Hierarchical depth aware YOLO for efficient metal surface defect detection," Scientific Reports, vol. 16, Article 16248, 2026, doi: 10.1038/s41598-026-46074-z.

[14] J. Tang, Q. Zhang, J. Jiang, Y. Pan, and X. Ding, "CPFS-YOLO: A steel surface defect detection method based on cross-stage and multi-scale feature fusion," Measurement Science and Technology, vol. 36, no. 11, p. 115402, 2025, doi: 10.1088/1361-6501/ae1854.

[15] S. Wang, L. Zhang, and G. Yang, "Defect identification method for steel surfaces based on improved YOLOv5," Journal of Southeast University (English Edition), vol. 40, no. 1, 2024.

[16] W. Zhao, F. Chen, H. Huang, D. Li, and W. Cheng, "A new steel defect detection algorithm based on deep learning," Computational Intelligence and Neuroscience, vol. 2021, Art. no. 5592878, 2021, doi: 10.1155/2021/5592878.

[17] Y. Tian, Q. Ye, and D. Doermann, "YOLOv12: Attention-Centric Real-Time Object Detectors," in Advances in Neural Information Processing Systems, vol. 38, pp. 78433–78457, 2025, doi: 10.52202/085713-2627.

[18] M. Yurdakul and A. Çakmak, "Multi-scale performance benchmarking of YOLO models for effervescent tablet defect detection," Preprints.org, 2026, doi: 10.20944/preprints202604 .1475.v1.

[19] D. Saif, H. Askr, A. M. Sarhan, and A. E. Hassanien, "Enhanced YOLO12 with spatial pyramid pooling for real-time cotton insect detection," Scientific Reports, 2026.

[20] X. Zhou and S. Agaian, "A review of YOLO-based object detection for e-waste sorting and recycling," in Proc. SPIE, vol. 14027, Jun. 2026, Art. no. 140270O.

[21] Ö. Akar, H. Selim, and O. Er, "Hyperparameter optimization for enhancing personal protective equipment (PPE) detection performance using YOLO-based deep learning," Intelligent Data Analysis, pp. 1–15, 2026, doi: 10.1177/1088467X26144573

[22] L. Ye, X. Chen, J. Lei, and X. Chen, "RFDNet-YOLOv13: A lightweight YOLOv13n-based model with receptive-field and dynamic head for robust tobacco leaf disease detection," Smart Agricultural Technology, p. 102069, 2026.

[23] N. Köklü and S. A. Sulak, "Classification of environmental attitudes with artificial intelligence algorithms," Intelligent Methods in Engineering Sciences, vol. 3, no. 2, pp. 54–62, 2024.

[24] L. T. Ramos and A. D. Sappa, "A decade of You Only Look Once (YOLO) for object detection: A review," IEEE Access, vol. 13, pp. 192747–192794, 2025, doi: 10.1109/ACCESS.2025.3597504.

[25] C. M. Liu and J. C. Juang, "Systematic evaluation of YOLOv8 variants for UAV-based object detection," Applied Sciences, vol. 16, no. 7, p. 3559, 2026.