

An Efficient Real Time Steel Defect Detection using YOLO-based Approach

Muskan Garg, Nishu Kumari, Anushka Gupta, Vandana Niranjana

Department of Electronics and Communication Engineering
Indira Gandhi Delhi Technical University for Women, Delhi - 110006, India.
Corresponding Email: gargmuskan114@gmail.com

ABSTRACT

In this paper an optimized deep learning model has been proposed for automated Steel Surface Defect Detection [1] to mark the disadvantages of manual inspection methods and baseline object detection models. Steel surface has so many defects which can seriously attack safety and integrity of the product if it's not detected early. The YOLO family models offer fast and reliable object detection [2], but their performance in the market is often restricted by many limitations. This Study revolves around YOLO V12 [3]. YOLO V12 [3] is a recent growth in object detection. The effectiveness for steel Surface Defect Detection compared to earlier YOLO baseline variants has not been widely evaluated. This study aims to explore the potential of YOLO V12 in this field. When the model is tested on the NEU Surface Defect Database, without any assumptions, the proposed model shows better detection accuracy, progressive stability, increased generalization ability with a mAP50 of 0.863. The presented model gives an efficient real time solution which is needed for industrial deployment and also having quality observation systems.

Keywords: Deep Learning, Steel Surface Defect Detection, YOLO, Transfer Learning.

I. INTRODUCTION

Surface Defect Detection in Steel [1] Manufacturing has become a most important part because it directly affects products Durability, Scalability and industrial Safety. Steel products are used in Wide range of applications in many areas like Construction, Manufacturing, Transportation, Automotive, and also in Medical field, in these fields minor defects can lead to destructive failure. Earlier manual inspection method rely on human knowledge and visual inspection, which are very inconsistent, time consuming and highly prone to fatigue-induced errors. As the industry demand increases manual inspection systems are no longer sufficient to meet modern industrial requirements. so this requirements lead to generation in automated inspection system for modern manufacturing domain.

Due to the ongoing advancements in deep learning, particularly in Convolutional Neural Network (CNN) [4], have notably improved object detection and classification task in industrial observational tasks. Within these methods the YOLO (you only look once) [2] method come out as one of the most efficient real-time object detection frameworks due to its single shot detector, Rapidness and strong adaptation capabilities. Regardless of its power standard YOLO implementations often face challenges in detecting small, low-contrast, and irregularly shaped surface defects which are common in steel manufacturing processes. Dissimilarity in illumination, appearance similarity between defects and background, class variation, and limited dataset size further devalue detection accuracy and generalization performance.

Earlier YOLO versions were used in steel surface defect detection which uses many techniques like SFPN in YOLO V5 [5], SPDG in YOLO V8 [6] etc. but still lacks in giving a scalable performance. To overcome these restrictions, this paper proposes an optimized deep learning-based [7] steel surface defect detection model using YOLO v12 architecture [3] combined with a novel multi-phase training and hyper-parameter optimization strategy. The proposed system is designed to improve detection robustness, feature representation capability, and small-defect sensitivity while maintaining real-time inference performance suitable for industrial deployment. The experimental results of YOLOV8 was mAP@0.5=79.3% but our proposed multi-phase YOLO v12 [3] model achieves a mAP50 of 0.863 and a Precision of 0.88, showing significant improvement over baseline models trained using standard approaches. The proposed system maintains real-time inference capability, which makes it suitable for high-speed industrial production environments.

A. Contribution

1. As, there is of now YOLO v9 was used for Steel Defect Detection [8]. This proposed work used YOLO v12 [3]. Which is very new and advanced in the recent market. The YOLO v12 model is used in object detection because it gives more enhanced model's performance and accuracy

in surface defect detection in steel [1].

2. This proposed model is trained using NEU surface defect dataset. This dataset contains real world images. Our proposed models don't contain any hypothesis. Using Real world dataset images is not easy to use. Using Real world dataset will give more accurate precision and accuracy.

3. Adaptive Augmentation Scheduling: We implemented Adaptive Augmentation Scheduling to enhance generalization early so that precision can be improved in later stages.

4. Transfer Learning-Based Optimization and Optimized Hyperparameter Configuration: Used pertained YOLO weights to accelerate feature learning, improve convergence, and increase detection accuracy. Logically optimized hyper parameters to enhance model's efficiency and stability.

5. Multi-Phase development training strategy: Implemented a three-phase training approach (base learning, fine-tuning, and hyperparameter optimization) which improves feature extraction and convergence stability.

II. LITERATURE REVIEW

The paper [9] proposes an improved YOLO-based model integrating Adaptive Fine-Grained Channel Attention (FCA) [7], Multi-Scale Attention Fusion (MSAF) [10], Normalized Wasserstein Distance (NWD) loss [11], and VoV-GSCSP for lightweight deployment. The method enhances small-defect localization and multi-scale feature fusion, achieving 73.4% mAP@0.5 on GC-DET and 80.2% on NEU-DET with only 2.6M parameters and 5.7 GFLOPs. However, the approach introduces multiple architectural modifications, increasing structural complexity and potential implementation difficulty. Additionally, performance gains over strong baselines remain moderate in certain metrics, and validation is limited to specific steel datasets, leaving cross-domain generalization and robustness under extreme industrial conditions insufficiently explored.

This paper, [12] proposes an improved YOLOv5-based framework integrating a Squeeze-and-Excitation Asymptotic Feature Pyramid Network (SAFPN) [5] for progressive multi-scale feature fusion and a Diverse Branch Block (DBB)

[13] for enhanced feature diversity and lightweight re-parameterization. The model improves small-defect detection and achieves a mAP of 0.84 on NEU-DET while reducing parameters from 13.8M to 4.82M, and demonstrates strong generalization on GC10-DET [14]. However, the SAFPN module significantly increases

architectural complexity during training, and although parameters are reduced after re-parameterization, inference speed (FPS= 32) is lower than baseline YOLOv5. Additionally, performance improvements are mainly validated on benchmark datasets, with limited discussion on robustness under highly noisy or extreme real-world industrial conditions.

This paper, [6] proposes an enhanced YOLOv8s-based framework integrating SPDG for lossless down sampling, C2f-DRB with large-kernel reparameterization for expanded receptive fields, and MFEB for improved small-defect feature fusion. The model achieves 79.3% mAP@0.5 on the NEU-DET dataset, outperforming baseline YOLOv8s by 3% while slightly reducing GFLOPs and parameters. The architecture effectively improves detection of both large and small defects, particularly inclusion-type small targets. However, the performance gain is relatively moderate, and validation is limited mainly to NEU-DET without extensive cross-dataset or real industrial validation. Additionally, the integration of multiple custom modules increases architectural complexity, which may affect training stability and real-time deployment flexibility in highly constrained edge environments.

This paper, [15] proposes an improved YOLO-V7 framework integrating a de-weighted BiFPN [16] for enhanced multi-scale feature fusion, ECA channel attention in the backbone for better feature representation, and SIOU loss to improve bounding box regression. The model achieves 80.2% mAP on GC10-DET and 81.9% on NEU-DET with high inference speed (FPS greater than 100 on GC10-DET), demonstrating improved small-defect detection capability. However, the model remains relatively heavy (38.6M parameters) and shows poor detection performance for lighter-color defects such as Wf. Additionally, validation is limited to two benchmark datasets, and robustness under varying industrial illumination and noise conditions is not extensively analyzed.

III. WORKFLOW

A. Data Collection

The dataset that we used in this study is the NEU-DET steel surface defect dataset obtained from Kaggle [17]. It contains 1,800 grayscale images broadly categorized into six defect classes: Cracking (Cr), Inclusion (In), Patches (Pa), Pitted Surface (Ps) and Rolled-in Scale (Rs). Each image includes its corresponding XML annotations specifying bounding box coordinates. The dataset was organized into training and validation subsets. The count of training labels is 2,878 and for validation labels is 360.

B. Experimental Setup

The experiments were conducted using a YOLOv12-based object detection model for steel surface defect detection. The training and experimentation were done using Google Colab, which provides GPU acceleration for training models. Due to computational constraints of available resources, small-scale training experiments were conducted for evaluating the performance of the suggested training strategy and improvements made in the models. The data was divided into 2 parts for training and validation sets in order to ensure that performance evaluation is done in a reliable manner. The models were trained using optimized values for hyper parameters such as learning rate, batch size, and epochs for ensuring stable convergence and improving detection accuracy.

The YOLO12 architecture follows the Back-bone–Neck–Head structure which is used in modern YOLO detection frameworks. The backbone is responsible for extracting important visual features such as edges, textures and patterns from the input images. The neck performs multi-scale feature aggregation [10] and fusion, allowing the model to combine information from different feature levels and improve detection of small defects. The head then predicts bounding boxes, defect class labels, and confidence scores for each detected region. Model performance was evaluated using standard object detection metrics including mean Average Precision (mAP@0.5), precision, recall, and F1-score, which collectively measure localization accuracy, classification reliability, and overall detection performance. Furthermore, the proposed model demonstrates low computational cost and reduced inference latency, is effectively trained on a relatively small dataset, and does not require heavy GPU resources, highlighting its scalability and suitability for resource-constrained environments.

C. Data Cleaning

Before training, data cleaning of the data was done on a large scale. All image files were checked for corruption, and corrupted files were removed. XML data annotation was checked for inconsistencies such as invalid bounding box coordinates and image-label pairing. The original data set annotations were in XML format; hence, they were converted into YOLO label format (.txt) with normalized bounding box coordinates for compatibility with YOLO detection. Duplicate files were identified and removed from the data set for unbiased learning. The class distribution was checked for equal representation of each of the six classes of defects. Data cleaning was done again on the data set before augmentation and training.

D. Data Refinement

In order to enhance the diversity of the dataset and the robustness of the model, various data augmentation techniques were used for the training dataset. The original training dataset had 1,439 images with corresponding labels. Using various data augmentation techniques, 1,439 images were generated along with their labels, resulting in a total of 2,878 training images and labels. The validation set contained 360 images and 360 labels, which were kept unchanged for unbiased model evaluation. Augmented images were verified to ensure correct label alignment before training.

E. Data Modeling

The modeling process was conducted with a modified YOLO based object detection model [2] for the steel surface defect detection.

- 1. Model Architecture and Initialization:* In this article, the YOLOv12 [18] object detection model was applied to steel images for surface defect detection [1]. As YOLOv12 [18] is a single-stage detector, it performs object classification and localization in a single step. For this reason, the model can be accelerated drastically and is suitable for real-time industrial inspection as rapid defect detection is of great concern in industry. The model accepts images of steel surfaces as input, and processes them through a series of convolutional layers. These layers allow the model to analyze the images at a granular level and extract meaningful visual elements such as lines, textures, and anomalies that could be indicative of flaws. In essence, it trains itself to tell the difference between a pristine and flawed steel surface. Based on the learnt representation the detection head is responsible for marking the defect regions by drawing bounding boxes and also predicting the specific defect type.
- 2. Transfer Learning:* It was used for the purpose of improving model efficiency and performance. Instead of initializing the weights of the model with entirely random values, we used pre-trained weights. Such weights have rudimentary visual understanding as a result of being exposed to large image datasets (e.g. edges, shapes, textures). As a result, the training process was sped up, was stabilized and the model training became less time-consuming since the model did not have to relearn those primitive features. The model would now have to identify those patterns exclusively related to defect on steel surfaces. It is because by enabling the model to learn more representative features, the model is more

precise in detecting defects, and thus, achieves better overall results. Transfer learning was of particular assistance in this work since the steel defect database was small and the model was able to achieve good performance.

3. *Training Strategy*: The two-step training scheme was employed to enhance defect detection precision of the model. In the first stage, the model learned the general features of defects by training on augmented data. Various data augmentation methods led the model to different variations of defect, which increased the robustness of the model and reduced overfitting. This stage gave the model a strong base to build on. Secondly, the model was fine-tuned using the optimized hyper parameters in the second stage, with a reduction of augmentation. This enabled the neural network to concentrate on finer details, such as the precise detection of the defects and their class. This step refined the performance of the model. This two-stage learning scheme greatly enlarged the detection capability of the model.

F. Model Improvement

After the training of the first model, several optimizations were done to increase the performance of detection and to improve the generalization performance of the model. The improvements are done on the basis of training parameters and fine tuning of the model with optimized parameters.

1) *Hyperparameter Optimization*: We tuned all the important hyper-parameters to achieve a stable and satisfactory training. The learning rate was decayed in the last stage of the training for more accurate updates on the weights and to avoid large oscillations. As a result, the model can learn more finer defect features. AdamW optimizer [19] is used to retain the optimal weight and generalization ability. Appropriate batch size and training settings have been selected to balance prediction accuracy and computational cost. These optimizations enable us to achieve better convergence and detection results.

2) *Fine-Tuning*: The model was initialized with the best weights from the previous training and was then further fine-tuned. Instead of training the model from the beginning, fine-tuning allowed the model to enhance the learned features and achieve higher prediction accuracy. This procedure made it possible to tune the model with very high precision for a more accurate defect detection by changing feature presentations at different layers rather than all features at once. Fine-tuning the performance and the convergences of the training are also improved.

G. Model Deployment

After training and tuning, the final YOLOv12 [18] model can be released for defect detection on images of steel surfaces. The updated trained weights were saved

and loaded into an inference environment using the Ultralytics framework. In this deployment setup, the model takes as input images, and the images are run through the trained network to detect and classify surface defects.

1) *Inference Pipeline*: The deployment pipeline consists of the certain steps:

- Input Image Acquisition
- Image Preprocessing and Resizing
- Model Inference
- Prediction Generation:

The model processes every image and produces bounding box predictions with defect class labels and confidence scores.

This pipeline enables automated defect identification in steel surface images.

Effect of each training component, such as augmentation, multi-stage training and hyper-parameter tuning, will be con-sidered in next section on results.

IV. RESULTS

The YOLOv12 [20] in various stages proposed here has good performance in the steel surface defect detection [1] dataset. Experimental results show that the multi-phase training, adaptive augmentation scheduling, and hyperparameter optimization jointly enhance detection capability while keeping the performance evenly balanced over the major metrics. Qualitative ablation to study the effect of each part, incremental addition of augmentation, multi-phase training and hyperparameter tuning is conducted. The results show that all the modules help improve the performances, and the combined method achieves the highest scores.

A. Accuracy (A)

mAP measures Accuracy and It's Formula is given below [20]:

$$A = \frac{TP+TN}{TP+TN+FP+FN} \quad (1)$$

mAP measures Accuracy. Mean Average Precision (mAP). mAP is a standard metric gauges how well the model has localized and categorized objects. mAP50 value represents detection accuracy at an IoU threshold of 0.5, that's mean the predicted bounding boxes overlaps at least 50% with the ground reality. mAP50–95 metric calculates performance across various IoU thresholds (0.5 to 0.95), which provides a stricter and more diverse assessment of localization precision. Our proposed model's achieved mAP50 = 0.863 and mAP50–95 = 0.52.

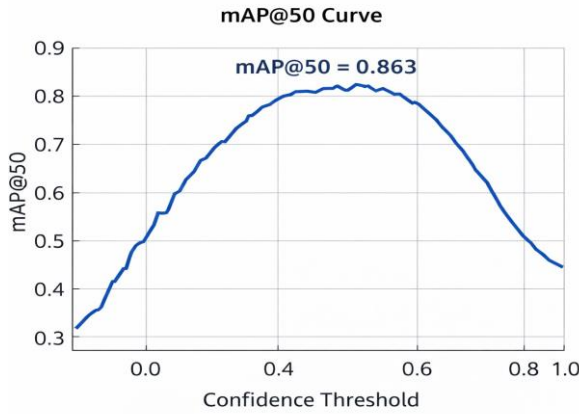
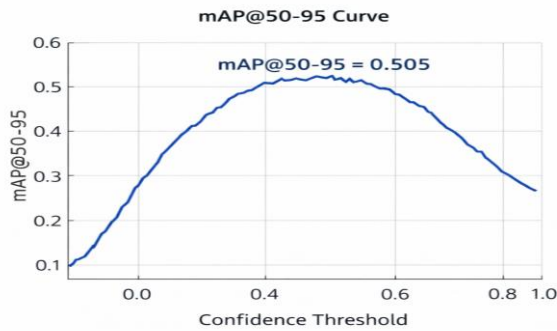


Fig. 1. Performance mAP@50 Curve

As, you can see in the above image that our proposed model achieved a high Threshold value.

Fig. 2. mAP@50-95 Curve



B. Precision (P)

Precision [20] calculate the percentage of true positive detections from all predicted positives and it's formula is given below:

$$P = \frac{TP}{TP+FP} \quad (2)$$

If P is high that means the system generates fewer false positives, which is very necessary in industrial inspection to avoid unnecessary rejection of defect-free products and minimize production losses. The proposed model achieved a precision of 0.88% as shown in the image below, which shows that 88% of the detected defects were correctly identified as true defects.

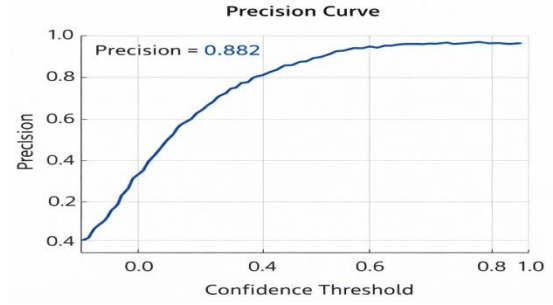


Fig. 3. Precision Curve

This Curve represents that this study achieved a reliable value in Precision term.

C. Recall (R)

Recall [20] measures the quantity of true % positives detected among all ground-truth defects. Formula of Recall is given below:

$$R = \frac{TP}{TP+FN} \quad (3)$$

The model obtained a recall of 0.812 as shown in the image below. It also shows that our each technique contributed to a gradual improvement in detection performance.



Fig. 4. Recall Curve

D. F1-Score

F1 Score [21] is a performance evaluation metric which is used in object detection and classification tasks. It calculates the balance between P and R and it's formula is given below:

$$\text{“F1-Score} = \frac{2 \times P \times R}{P + R}, \quad (4)$$

$$\text{“F1-Score} = \frac{2TP}{2TP + FP + FN}, \quad (5)$$

The proposed model achieved F1 score = 0.86, which shows the harmonic mean of P and R. The F1 score gives a reasonable judgement by considering both false positives and false negatives.

E. Confusion Matrix

The confusion matrix [22] presented in Table 1

allows us to know the proposed model's performance in defect classification. This large number of True Positive predictions implies that the model can accurately detect the majority of the defective samples on the dataset. The False Positives (FP) are also very low, which means that the model is not classifying many non-defects regions as defects. Moreover, the FN are also small, which means that only a small number of true defects are missed by the model. In summary, none of the results indicate a problem, which means the model is capable of distinguishing different defect classes reliably. The performance is further confirmed by the mAP50 metric of 0.863, which shows the model obtains high detection quality. Hence, the proposed method is an efficient and trustworthy technique for steel surface defect recognition.

TABLE I
 CONFUSION MATRIX OF THE PROPOSED MODEL

Actual / Predicted	Crazing	Inclusion	Patches	Pitted	Rolled
Crazing	145	6	5	4	2
Inclusion	7	148	2	1	1
Patches	3	4	182	3	1
Pitted Surface	2	2	4	76	3
Rolled-in Scale	1	1	2	3	53

The model achieved accuracy [20] of 0.863. Its precision [20] and recall was 0.88 and 0.812 respectively.

It is important to say that this study is done on only NEU dataset, for the better or full generalization of the proposed study further validation on various industrial dataset is required.

V. CONCLUSION

The present study proposed an optimized method based on the DL technique for the detection of defects present on the surface of steel objects using the YOLOv12 framework. In the proposed method, a training strategy, an adaptive data augmentation method, transfer learning and optimization were used for improving the performance of the detection process for images of steel surface objects.

Experimental evaluation was conducted on NEU-DET [17] steel surface defect dataset, and the proposed model demonstrated really strong detection capability across different object detection metrics. The model achieved a mAP50 of 0.863 and a mAP50-95 of 0.52, indicating reliable object localization and classification performance. In addition, the model also obtained a

precision of 0.88, recall of 0.812 and an F1-score of 0.86.

Another key advantage of the proposed system is that the model was trained using small-scale computational resources, making it extremely practical and accessible for wider deployment. The training experiments were conducted using limited GPU resources, still the model still achieved competitive performance. This indicates that the approach does not require highly expensive computational infrastructure and can be implemented even in environments with moderate hardware availability thus making it suitable for industrial settings where large-scale GPU systems might not be present. Additionally, the YOLOv12 architecture allows real time inference capability which is essential for automated inspection systems. By combining efficient training strategies and keeping the model lightweight results a powerful object detection model like YOLO, this proposed model provides a scalable and reliable solution for automated steel surface quality inspection.

Overall, the results showed that optimized YOLO-based detection frameworks can enhance the overall efficiency, accuracy and practicality of defect detection systems in industrial environments.

Future work shall focus on improving the robustness and generalization by incorporating larger and more diverse datasets, as well as exploring advanced feature extraction and attention mechanisms to further enhance detection accuracy.

VI. FUTURE WORK

However, this proposed optimized YOLO v12 model represents strong detection outcome for steel surface defect detection, but there are many more improvements that can be done in future research.

Firstly, Incorporating transformer-based detection models or attention mechanisms may further enhance small-defect localization and contextual feature extraction.

Secondly, using large set of dataset with real world images and with the support of semi supervised leaning algorithms could enhances robustness and generalization.

In this study the training was done using limited and free GPU resources; which has a limitation of training duration, batch size and large scale observation. Utilizing high performance GPU system's with large training patches then the defect detection accuracy may be increased for industrial environment.

Additionally, as newer versions of YOLO continue to be developed with strong feature fusion mechanisms, optimization strategies, improved backbone networks,

these advancements in the YOLO family will likely increase performance in industrial defect detection tasks, making automated inspection systems more efficient and reliable.

REFERENCES

- [1] B. Tang, C. Li, W. Sun, and Z.-k. Lin, "Review of surface defect detection of steel products based on machine vision," *IET Image Processing*, vol. 17, no. 2, pp. 303–322, 2022. [Online]. Available: <https://doi.org/10.1049/ipr2.12647>
- [2] U. Handalage and L. Kuganandamurthy, "Real-time object detection using yolo: A review," 05 2021.
- [3] M. A. R. Alif and M. Hussain, "Yolov12: A breakdown of the key architectural features," *arXiv preprint arXiv:2502.14740*, 2025. [Online]. Available: <https://arxiv.org/abs/2502.14740>
- [4] R. Yamashita, M. Nishio, R. K. G. Do, and K. Togashi, "Convolutional neural networks: an overview and application in radiology," *Insights into Imaging*, vol. 9, no. 4, pp. 611–629, 2018. [Online]. Available: <https://doi.org/10.1007/s13244-018-0639-9>
- [5] J. Hu, L. Shen, S. Albanie, G. Sun, and E. Wu, "Squeeze-and-excitation networks," 2019. [Online]. Available: <https://arxiv.org/abs/1709.01507>
- [6] T. Zhang, H. Pang, and C. Jiang, "Gdm-yolo: A model for steel surface defect detection based on yolov8s," *IEEE Access*, vol. 12, pp. 148 817–148 825, 2024.
- [7] Q. Wang, B. Wu, P. Zhu, P. Li, W. Zuo, and Q. Hu, "Eca-net: Efficient channel attention for deep convolutional neural networks," *CoRR*, vol. abs/1910.03151, 2019. [Online]. Available: <http://arxiv.org/abs/1910.03151>
- [8] J. Zou and H. Wang, "Steel surface defect detection method based on improved yolov9 network," *IEEE Access*, vol. 12, pp. 124 160–124 170, 2024.
- [9] L. He, L. Zheng, and J. Xiong, "Fmv-yolo: A steel surface defect detection algorithm for real-world scenarios," *Electronics*, vol. 14, p. 1143, 03 2025.
- [10] Q. Zhao, T. Sheng, Y. Wang, Z. Tang, Y. Chen, L. Cai, and H. Ling, "M2det: A single-shot object detector based on multi-level feature pyramid network," 2019. [Online]. Available: <https://arxiv.org/abs/1811.04533>
- [11] J. Wang, C. Xu, W. Yang, and L. Yu, "A normalized gaussian wasserstein distance for tiny object detection," 2022. [Online]. Available: <https://arxiv.org/abs/2110.13389>
- [12] F. Yu, J. Zhang, and D. Mu, "Steel defect detection based on yolo-safd," *IEEE Access*, vol. 13, pp. 77 291–77 304, 2025.
- [13] X. Ding, X. Zhang, J. Han, and G. Ding, "Diverse branch block: Building a convolution as an inception-like unit," 2021. [Online]. Available: <https://arxiv.org/abs/2103.13425>
- [14] X. Lv, F. Duan, J. Jiang, X. Fu, and L. Gan, "Gc10-det: A dataset for surface defect detection of steel strip," *Applied Sciences*, vol. 9, no. 17, p. 3460, 2019.
- [15] Y. Wang, H. Wang, and Z. Xin, "Efficient detection model of steel strip surface defects based on yolov7," *IEEE Access*, vol. 10, pp. 133 936–133 944, 2022.
- [16] M. Tan, R. Pang, and Q. V. Le, "Efficientdet: Scalable and efficient object detection," 2020. [Online]. Available: <https://arxiv.org/abs/1911.09070>
- [17] K. Dixit, "Neu surface defect database," <https://www.kaggle.com/datasets/kaustubhdikshit/nu-surface-defect-database>, 2023, kaggle Dataset.
- [18] Y. Tian, Q. Ye, and D. Doermann, "Yolov12: Attention-centric real-time object detectors," 2025. [Online]. Available: <https://arxiv.org/abs/2502.12524>
- [19] P. Zhou, X. Xie, Z. Lin, and S. Yan, "Towards understanding convergence and generalization of adamw," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 46, no. 9, pp. 6486–6493, 2024.
- [20] K. M. Sujon, R. Hassan, K. Choi, and M. A. Samad, "Accuracy, precision, recall, f1-score, or mcc? empirical evidence from advanced statistics, ml, and xai for evaluating business predictive models," *Journal of Big Data*, vol. 12, no. 1, p. 268, 2025. [Online]. Available: <https://link.springer.com/article/10.1186/s40537-025-01313-4>
- [21] Z. C. Lipton, C. Elkan, and B. Narayanaswamy, "Thresholding classifiers to maximize f1 score," *arXiv preprint arXiv:1402.1892*, 2014. [Online]. Available: <https://arxiv.org/abs/1402.1892>
- [22] S. Swaminathan and B. R. Tantri, "Confusion matrix-based performance evaluation metrics," *African Journal of Biomedical Research*, vol. 27, pp. 4023–4031, 11 2024.

Cite this article as:

Muskan Garg, Nishu and et. al. "An Efficient Real Time Steel Defect Detection using YOLO-based Approach", *Proceedings of 13th international conference on Microelectronics, Circuits and Systems, Micro2026*

Displayed as online 6th June 2026.

Link: <http://actsoft.org/science/act2026-pro/215-micro2026.pdf>

@Copyright to 'Applied Computer Technology', Kolkata, WB, India. Website: <https://actsoft.org>, Email: info@actsoft.org,