International Refereed Journal of Engineering and Science (IRJES)
ISSN (Online) 2319-183X, (Print) 2319-1821
Volume 15, Issue 3 (May. – June. 2026), PP. 36-46

Classification and Explainable Performance Analysis of Industrial Surface Defects Using Deep Learning-Based CNN and Transformer Architecture

Büşra ER[a,*], Canan TAŞTİMUR[b], Volkan KAYA[c]
[a]*M.Tech student, Erzincan Binali Yıldırım University, Graduate School of Natural and Applied Sciences, Department of Artificial Intelligence and Robotics, Erzincan, Türkiye
[b]Asst. Prof. Dr. Erzincan Binali Yıldırım University, Faculty of Engineering and Architecture, Departmant of Computer Engineering, Erzincan, Türkiye,
[c]Assoc. Prof. Dr. Erzincan, Türkiye Erzincan Binali Yıldırım University, Faculty of Engineering and Architecture, Departmant of Computer Engineering, Erzincan, Türkiye
Corresponding author: Büşra ER

Abstract:*This study investigates the problem of classifying surface defects, considered important in quality control in industrial production processes, using deep learning-based methods. Accordingly, a comprehensive performance analysis and comparison was carried out on different methods based on Convolutional Neural Networks (CNN) and Transformer. Experimental studies were conducted on the NEU Surface Defect dataset, and both the hold-out test method and the 5-fold cross-validation (CV) strategy were used to more reliably evaluate the generalization performance of the models. Model performance evaluation was performed using accuracy, loss, and macro F1-score metrics, and the results were presented with mean and standard deviation values. Furthermore, to make the models' decision-making processes more understandable, Grad-CAM-based visualizations were used to examine the networks' behavior in detail, focusing on flawed regions. The results show that Transformer-based methods offer more stable and balanced generalization performance in some cases, while CNN-based methods can achieve lower computational cost.*

Keywords:*Industrial surface defects, deep learning, CNN and Transformer models, Grad-CAM*

Date of Submission: 25-05-2026 Date of acceptance: 05-06-2026

I. Introduction

In industrial production processes, the rapid and accurate detection of surface defects is of great importance in both maintaining product quality and reducing production costs, in order to ensure customer satisfaction. Defects on metal surfaces, in particular, directly affect the aesthetic structure, durability and overall performance of the product, and are therefore among the important factors in terms of customer satisfaction. In traditional detection methods, each product needs to be examined individually by a human, which increases the risk of error depending on human factors such as experience, attention level, fatigue and working speed. Therefore, traditional inspection approaches are often insufficient in today's high-precision industrial applications [1].

Recently, the hybrid use of deep learning and computer vision methods has become quite widespread. These hybrid methods are particularly encountered in fault detection and diagnosis applications, yielding remarkable results. It has been observed that these approaches provide more successful performance compared to traditional machine learning and classical image processing methods, both in extracting the features of faulty areas and in classifying the detected faulty areas [2, 3]. For the diagnosis and detection of different defects occurring on industrial metal surfaces, the NEU dataset, which includes six different defect types, is referenced in the literature [4].CNN tabanlı yöntemlere alternatif olarak geliştirilen Vision Transformer ve Swin Transformer-Tiny miamriler de bulunamkatadır [5, 6].This study presents comparative results of CNN and transformer-based architectures, adhering to the same experimental protocol.

This study’s main contributions can be summarized as follows:

- A comprehensive performance comparison of CNN-based (EfficientNetV2-S, ConvNeXt-Tiny), Transformer-based (ViT-Small, Swin Transformer-Tiny, MobileViT-V2), and hybrid (CoAtNet-0) architectures is presented on the NEU Surface Defect dataset.

- In experimental studies, K-fold cross-validation was applied to ensure model generalization. The dataset used was split into 70/30.
- Model complexity, explainability analysis, and examination of results were performed.

The rest of the article is structured as follows: The second section is a literature review on the diagnosis of medial surface defects; the third section provides information about the deep learning modules used in the proposed method and the dataset used; the fourth section presents the comparative experimental results obtained; the fifth section discusses the limitations of the study; and the final section discusses the general conclusions.

II. Related Works

Various solutions have been produced in the literature for the classification of metal surface defects using CNN-based and transformer-based approaches. Zhu et al., proposed a Swin Transformer-based model and achieved significant performance improvements in the detection of steel surface defects, demonstrating that the model can better represent particularly complex defect structures [7]. Similarly, a Swin Transformer-based approach proposed by Tang et al., provided strong feature extraction in the detection of steel strip surface defects, achieving higher accuracy values compared to traditional CNN-based methods [8].Recent research by Assad et al. has also focused on developing hybrid approaches that combine the strengths of CNN and Transformer architectures. Such models combine CNN-based local feature learning with Transformer-based global modeling. Researchers have focused on developing hybrid approaches that combine the strengths of CNN and Transformer architectures in recent years. These models aim to provide more powerful representation learning by combining CNN-based local feature learning with Transformer-based global context modeling. Studies show that hybrid CNN–Transformer models can provide higher accuracy, especially in detecting small and complex defects [9].

On the other hand, Song's study aimed to detect surface defects using object detection-based approaches. For example, YOLO-based methods offer high speed and competitive accuracy values for real-time defect detection applications. In recent years, it has been reported that improved YOLO-based models can more successfully detect small and low-contrast defects on the NEU dataset [10]. Similarly, in more recent studies, Zhou et al. addressed the multi-scale defect detection problem using Transformer-based DETR architectures and achieved performance improvements, especially in the detection of small defects [11].Li et al. proposed a deep learning model based on multi-scale feature extraction for steel surface defect detection. The proposed model uses multilayer feature fusion mechanisms to more effectively represent defect patterns at different scales. Experiments have shown that multi-scale feature extraction improves the accuracy of surface defect classification [12].He et al. proposed a deep CNN-based approach for classifying steel surface defects. In the study, textural features of surface defects were learned using a multilayer convolutional structure, and high accuracy values were obtained on the NEU dataset. The researchers showed that deep CNN architectures are particularly effective in capturing the microstructural and textural features of surface defects [13].

In this study, in order to fill this gap in the literature, both CNN-based and Transformer-based modern architectures were comprehensively compared within the same experimental framework. The performance of the models was evaluated on the NEU Surface Defect dataset, and both hold-out testing and 5-fold cross-validation were used to more reliably analyze their generalization ability. Furthermore, not only performance metrics but also Grad-CAM-based explainability analyses were used to examine the models' behavior in focusing on defect regions.

III. Materials and Methods

3.1. Dataset

In this study, the NEU Surface Defect open-access dataset was used to classify industrial surface defects using deep learning-based methods[14].The NEU dataset covers of original grayscale surface images with a resolution of 200 × 200 pixels with six different types of surface defects. The numerical distribution of images for each defect class is given in Table 1. Additionally, sample surface images for each class are shown in Figure 1 to better visualize the defect types included in the dataset.

Table 1. Number of images belonging to the surface defect classes included in the dataset.

Class Name	Number of Images (#)
Crazing	240
Inclusion	240
Patches	240
Pitted surface	240
Rolled-in scale	240
Scratches	240

Class Name	Number of Images (#)
Total	1440

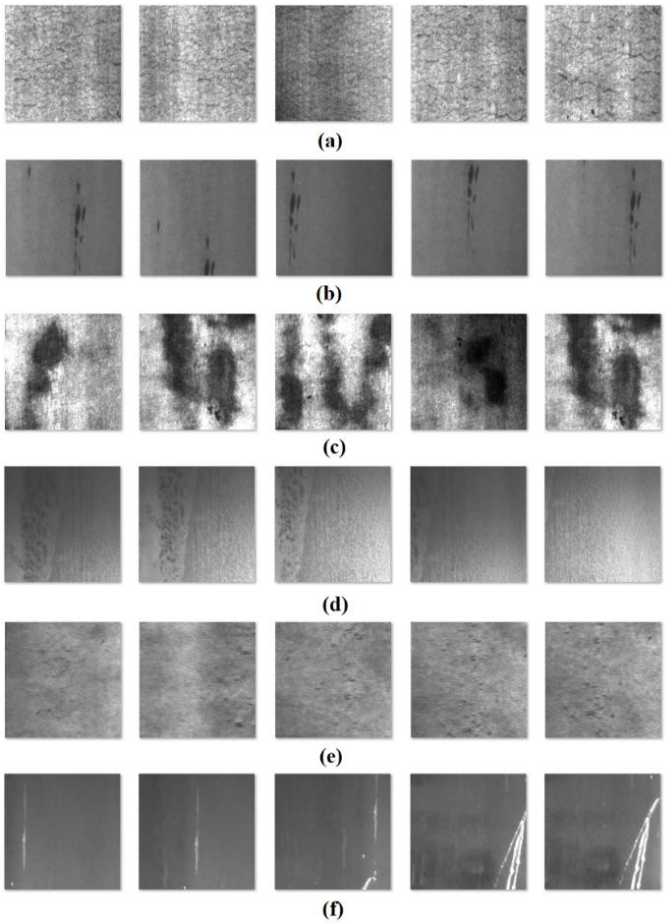


Fig. 1.Sample images of surface defect classes included in the NEU Surface Defect dataset.(a) Crazing (b) Inclusion (c) Patches (d) Pitted surface (e) Rolled-in scale (f) Scratches

Each class contains approximately a similar number of samples, and the dataset has a balanced distribution among the classes. This improves the suitability of the dataset for multi-class supervised learning problems. The 70% of the data was used as the training set, 15% as the validation set, and 15% as the test set. The training set was used for the model's learning process, the validation set for hyperparameter tuning and the early stopping mechanism, and the test set solely for final performance evaluation.

3.2. Pre-processing and Augmentation

All images in the dataset used for training deep learning-based models were subjected to a standard preprocessing process. First, the dimensions of the input images, which had a resolution of 200 × 200, were resized to 224 × 224 pixels to match the ImageNet pre-trained models. This process ensures that the model inputs have a consistent size, allowing for a more stable training process.

In order to both improve the generalization ability of the model and reduce overfitting during the training process, data augmentation techniques were applied only to the training set. Data augmentation techniques included random resized crop, random horizontal and vertical flip, random rotation, color jitter (brightness), and random erasure. These data enhancement methods aim to realistically reflect the different orientations, positional changes, and local textural variations frequently encountered in industrial surface defects.

3.3. Deep Learning Models

In this study, ConvNeXt-Tiny [15], EfficientNetV2-S [16], Swin Transformer-Tiny [6], CoAtNet-0 [17], Vision Transformer (ViT-Small) [5] and MobileViT-V2 [18] deep learning models with different architectural designs and using pre-trained weights on the ImageNet dataset are compared for the purpose of classifying industrial surface defects. Each of the models used in this study was carefully selected to represent different deep learning architectures. Models such as ConvNeXt and EfficientNetV2-S are based on

convolutional neural networks, while Vision Transformer and Swin Transformer-Tiny models are based on transformer-based architectures that rely on self-attention mechanisms. MobileViT-V2 and CoAtNet-0 stand out as hybrid architectures combining convolutional and transformer-based approaches. Each model structure was trained using the same dataset sets, with the same preprocessing and data enhancement strategies, and performance comparisons were made fairly and consistently.

3.4. Training Strategy

All deep learning models included in this study were trained using a transfer learning approach. Thus, all models were initialized with pre-trained weights on the ImageNet dataset and adapted to the task of classifying industrial surface defects. The preference for pre-trained weights for model training contributes to faster convergence and more stable training progress in applications with limited data volumes. During the model adaptation phase, the classification layer of each model was reconfigured to include the six defect classes involved in the problem. During training, all layers were updated and fine-tuned to ensure the model learned both low-level visual features and high-level representations task-specifically. To ensure fair comparison of all model experiments, the hyperparameters used in the training process were determined in the same way. Therefore, all models were trained using the AdamW optimization algorithm with a learning rate of 3×10^{-4} and a weight reduction coefficient of 0.05, and a Cosine Annealing-based learning rate timer was used to gradually reduce the learning rate throughout the training

In the training process of the models, cross-entropy loss, which is commonly preferred in multi-class classification problems, was used as the loss function.Label smoothing (0.1) was applied to the loss function to prevent the model from producing overly confident predictions and to improve generalization performance.The training process was limited to a maximum of 30 epochs. To prevent overfitting and reduce unnecessary training cycles, an early stopping mechanism was used based on the macro-average F1 score observed on the validation set. If no improvement in validation performance was observed, the training process was terminated after 3 epochs.This training strategy aims to ensure that all models are evaluated under the same conditions, making the obtained performance results comparable and reliable. Furthermore, the data processing, model training, and evaluation process followed in the classification of surface defects is schematically shown in Figure 2.

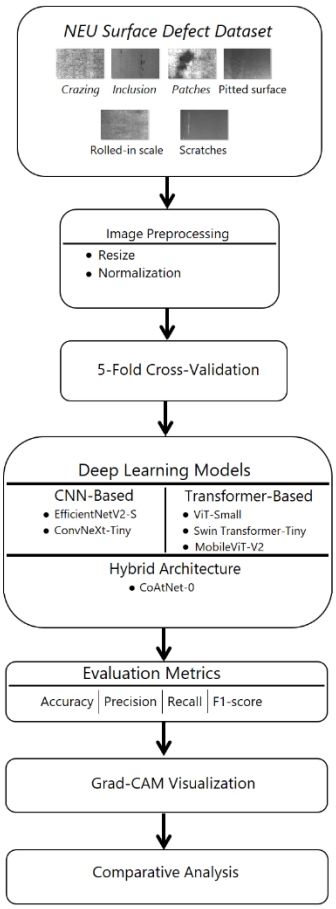


Fig. 2. Proposed experimental workflow for classifying surface defects.

3.5. Evaluation Metrics

In this study, the performance of deep learning models was analyzed using commonly used evaluation metrics suitable for the multi-class classification problem. Both single-compartment test set results and statistics obtained within the scope of five-fold cross-validation were considered in the evaluation process.The accuracy metric was used to measure the overall accuracy of the model performance. Accuracy expresses the ratio of correctly classified samples to the total number of samples. However, the accuracy metric alone was not considered sufficient for a more detailed examination of inter-class confusion.

Therefore, macro-average precision and macro-average recall metrics were reported by taking the macro-average of the precision and recall values calculated for each class. The macro-average approach provides a performance evaluation independent of inter-class imbalance by giving equal weight to each class.For a balanced measurement of model performance, the macro-average F1 score was used as the primary evaluation metric. The F1 score represents the harmonic mean of precision and recall metrics and provides an effective measure for evaluating generalization performance, especially in multi-class problems. In this study, the early stopping mechanism was also implemented based on the macro-average F1 score on the validation set.

To analyze the confusion between classes in detail, a confusion matrix was created for each model. Confusion matrices allow for the interpretation of model behavior by visualizing which types of defects are more frequently confused with each other.In five-fold cross-validation experiments, the evaluation metrics obtained for each fold were calculated separately, and the results were reported as mean ± standard deviation (mean ± std). The primary goal of this approach is to demonstrate the performance stability and generalization ability of models more consistently and reliably across different data subsets.

IV. Results

4.1. Accuracy and F1-score comparison

In this section, the deep learning models discussed are comparatively examined, considering key performance metrics such as classification performance, number of parameters, accuracy, macro precision, macro recall, and macro F1-score. All models were trained and tested using the same training strategy and datasets to ensure fair evaluation. Accuracy and macro mean F1-score values were calculated from the independent test set.To more robustly demonstrate the generalization ability of the models, a 5-fold cross-validation (CV) method was preferred, and both independent test set results and cross-validation findings were considered together in the evaluation process. All performance analyses were performed using macro mean metrics.In the cross-validation phase, the dataset was divided into five equal parts. In each iteration, four of these parts were used for training, while the remaining part was evaluated for validation. This process was repeated a total of five times to ensure that each subset was included as a validation set once. In each layer, the models were retrained with the same training procedure, and performance metrics were calculated on the relevant validation set. The results are presented in mean ± standard deviation (mean ± std) format to demonstrate the consistency of model performance (Table 2).

Table 2. 5-Fold Cross-Validation Results

Model	Number of Parameters	Accuracy (Mean ± Std)	Precision (Macro)	Recall (Macro)	F1-score (Macro)
ConvNeXt-Tiny	27.82 M	0.9885 ± 0.0054	0.9886 ± 0.0054	0.9885 ± 0.0054	0.9886 ± 0.0054
EfficientNetV2-S	22.16 M	0.9992 ± 0.0018	0.9992 ± 0.0018	0.9992 ± 0.0018	0.9992 ± 0.0018
Swin Transformer-Tiny	27.52 M	0.9861 ± 0.0170	0.9862 ± 0.0168	0.9862 ± 0.0168	0.9862 ± 0.0168
CoAtNet-0	26.67 M	0.9869 ± 0.0124	0.9869 ± 0.0124	0.9869 ± 0.0124	0.9869 ± 0.0124
ViT-Small	21.67 M	0.9877 ± 0.0077	0.9877 ± 0.0077	0.9877 ± 0.0077	0.9877 ± 0.0077
MobileViT-V2	1.12 M	1.0000 ± 0.0000	1.0000 ± 0.0000	1.0000 ± 0.0000	1.0000 ± 0.0000

According to the results presented in Table 2, some models consistently demonstrated high performance across all layers, while others showed greater performance differences between layers.

Table 3. Comparative evaluation of models using a test dataset

Model	Number of Parameters	Accuracy	Precision (Macro)	Recall (Macro)	F1-score (Macro)
ConvNeXt-Tiny	27.82 M	0.9908	0.9908	0.9907	0.9907
EfficientNetV2-S	22.16 M	**1.0000**	**1.0000**	**1.0000**	**1.0000**
Swin Transformer-Tiny	27.52 M	**1.0000**	**1.0000**	**1.0000**	**1.0000**
CoAtNet-0	26.67 M	0.9908	0.9910	0.9909	0.9909
ViT-Small	21.67 M	**1.0000**	**1.0000**	**1.0000**	**1.0000**
MobileViT-V2	**1.12 M**	**1.0000**	**1.0000**	**1.0000**	**1.0000**

A detailed examination of the results in Table 3 reveals that all models generally perform well; however, the EfficientNetV2-S, Swin Transformer-Tiny, ViT-Small, and MobileViT-V2 models achieved flawless classification success in the test dataset.MobileViT-V2, with its remarkable 1.12 million parameters, demonstrates performance comparable to the largest models.

4.2. Training - Validation Curves

To evaluate the stability and generalization ability of the model training process, training and validation loss and accuracy curves were analyzed for each model.

Throughout the training process:

- Training and validation loss values decreased in parallel,
- No sudden increases or unstable fluctuations were observed in validation loss,
- Validation accuracy remained close to training accuracy.

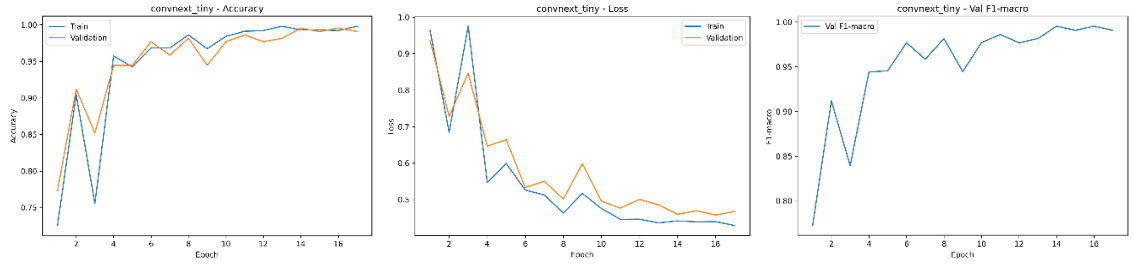
This situation shows that the models not only memorize the training data, but also learn representations with high generalization capabilities. Furthermore, with the activation of the early stopping mechanism, the training process was automatically terminated in epochs where a decrease in validation performance was observed. This strategy prevented overlearning of the models, resulting in more reliable results.

In the training process, short-term increases in training loss values were observed in some epochs. This situation stems from factors such as:

- CosineAnnealing learning rate timing,
- Randomness of data augmentation strategies,
- Use of AMP (Automatic Mixed Precision).

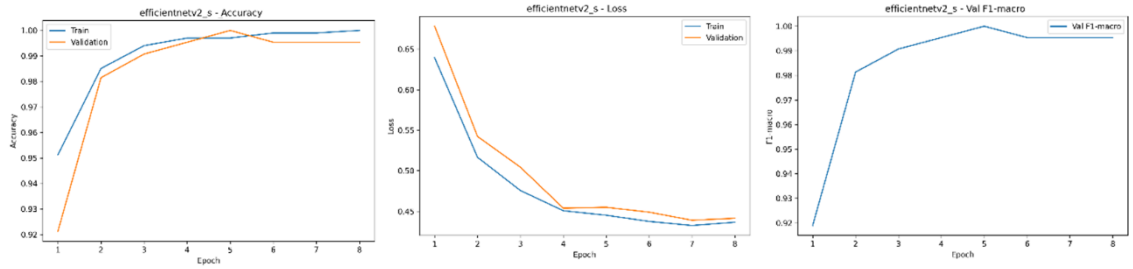
However, it was observed that these increases did not affect validation performance, and therefore did not have a negative impact on model stability.Training and validation accuracy, loss, and validation F1-macro curves for all models are presented in Figures 3–8. These graphs were used to evaluate the learning dynamics, convergence behavior, and generalization capabilities of the models.Throughout the training process, it was observed that the best-performing models showed progression in validation accuracy and F1-macro scores parallel to the training curves. This indicates that these models not only memorized the training data but also exhibited consistent and stable learning on the validation data. In particular, the stabilization and lack of increase in validation loss after a certain epoch supports the absence of overfitting.

CNN-based models (Figures 3 and 4) showed a rapid decrease in training loss and achieved high accuracy values in the early epochs. However, a limited difference between validation accuracy and training accuracy was observed in some CNN models. This difference may be attributed to the models focusing on more local features and representing the global context in a limited way.In transformer-based models (Figures 5 and 7), a slower convergence was observed in the initial epochs of the training process; however, validation accuracy and F1-macro scores increased steadily in the later epochs. Specifically, the low-volatility rise of the validation F1 curves indicates that these models have learned better inter-class equilibrium and exhibit consistent performance even on minority classes.A combined evaluation of validation accuracy, validation loss, and validation F1-macro curves for all models reveals that the best results are obtained by models with not only high accuracy values but also low and stable validation loss.



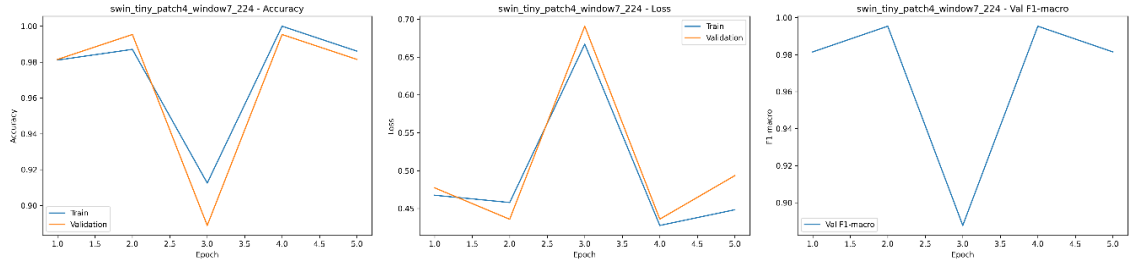
(a) **(b)** **(c)**

Fig. 3. (a) Training and validation accuracy, (b) training and validation loss, and (c) validation macro F1-score values for ConvNeXt-Tiny.



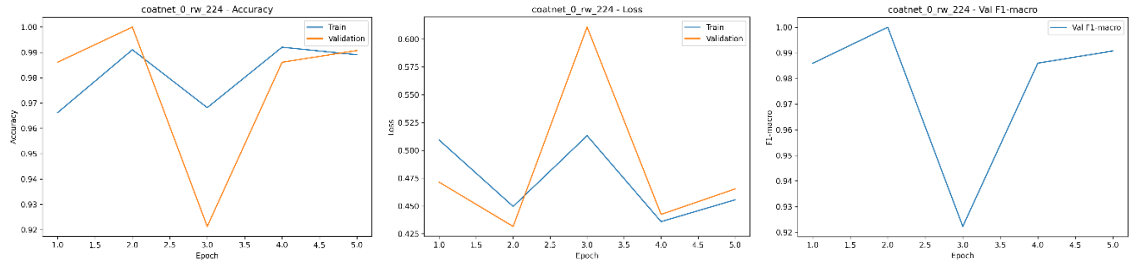
(a) **(b)** **(c)**

Fig. 4. (a) Training and validation accuracy, (b) training and validation loss, and (c) validation macro F1-score values for EfficientNetV2-S.



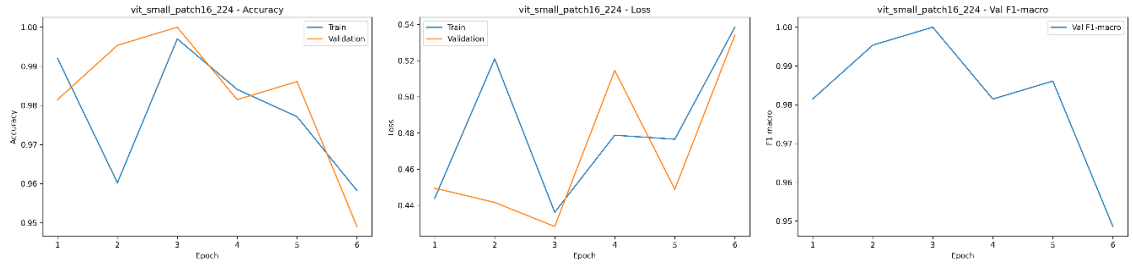
(a) **(b)** **(c)**

Fig. 5. (a) Training and validation accuracy, (b) training and validation loss, and (c) validation macro F1-score values for Swin Transformer-Tiny.



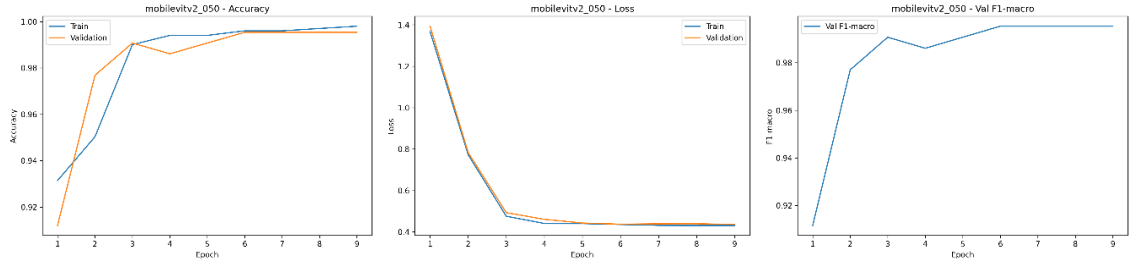
(a) **(b)** **(c)**

Fig. 6. (a) Training and validation accuracy, (b) training and validation loss, and (c) validation macro F1-score values for CoAtNet-0.



(a) **(b)** **(c)**

Fig. 7. (a) Training and validation accuracy, (b) training and validation loss, and (c) validation macro F1-score values for ViT-Small.



(a) **(b)** **(c)**

Fig. 8. (a) Training and validation accuracy, (b) training and validation loss, and (c) validation macro F1-score values for MobileViT-V2.

In this study, to evaluate the learning behavior of the models and the stability of the training process, learning curves for training-validation accuracy, loss, and validation F1-macro scores are presented for all models. These graphs are a sufficient and widely accepted assessment tool for visually analyzing whether the models exhibit overfitting or underfitting. The close and stable progression of the training and validation curves indicates that the models have learned the data distribution consistently.In comparing model performances, 5-fold cross-validation results were used; however, no single "best" fold was selected. Instead, validation performance for each model was reported using mean and standard deviation (mean ± std) values. This approach offers a more reliable assessment by demonstrating that the performance achieved is not specific to a single dataset, but is consistently obtained across different subsets.

To thoroughly analyze classification errors, the confusion matrices of all models were evaluated only on a separate test dataset. Because the test dataset used was completely independent of the training and validation phases, the resulting matrices demonstrate the accuracy of the models' performance under real-world conditions.In the case of cross-validation results (mean ± standard deviation) shown in Table 2, separate presentation of fold-based confusion matrices was not necessary. This choice ensures a simpler, comparable, and reproducible presentation of the results.Although the best model achieved 100% accuracy on the test dataset, this result is not indicative of overfitting. Firstly, no data leakage was detected between the training, validation, and testing phases. Secondly, the consistency between cross-validation and test dataset performances demonstrates strong generalization capability. Third, as shown in Figure 8, the training and validation curves exhibit stable convergence without divergence. Finally, the confusion matrices reveal a balanced classification across all defect categories, rather than rote memorization of dominant classes.

4.3. Confusion Matrix

Confusion matrices obtained from hold-out test results reveal the models' ability to discriminate between classes in detail. Overall, extremely low rates of misclassification between classes were observed across all models.ConvNeXt-Tiny and CoAtNet-0 models showed a limited number of errors, primarily occurring between the crazing-patches and inclusion-scratches classes. This is due to the fact that these defect types share partially similar visual characteristics in terms of surface texture.

In contrast, the confusion matrices of EfficientNetV2-S, Swin Transformer-Tiny, ViT-Small, and MobileViT-V2 models showed a full diagonal structure. This result indicates that these models were able to distinguish all defect classes in the test set without errors. The absence of false positives or false negatives explains the 100% success rate of macro-averaged metrics.When the results are examined, it is seen that not only the overall performance metrics but also the class-based discrimination success is extremely high. To maintain the flow of the study and avoid duplication, only the confusion matrix of the highest performing models is presented (Figures 9-10).

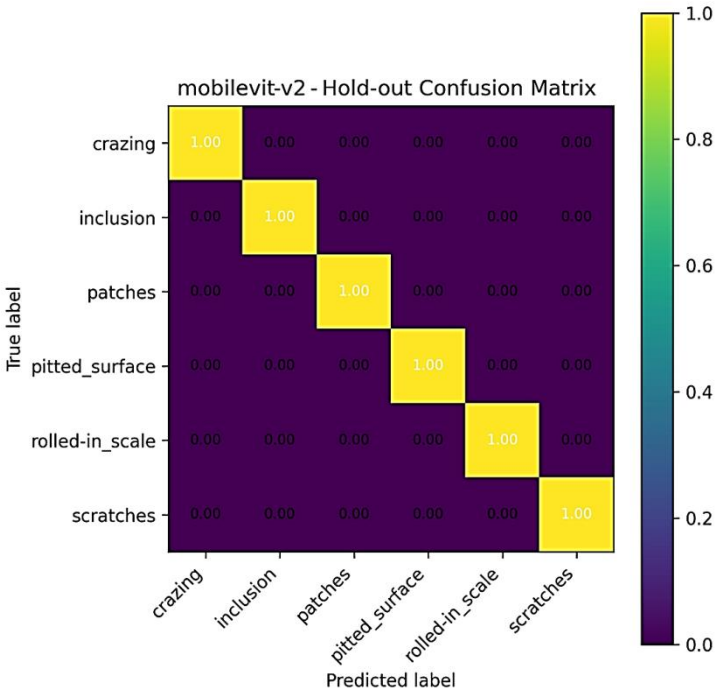


Fig. 9. Normalized confusion matrix of the MobileViT-V2 model on the hold-out dataset

Examining the confusion matrix in Figure 2, it can be seen that all error classes are classified with high accuracy, and that there is a balanced performance among the classes.

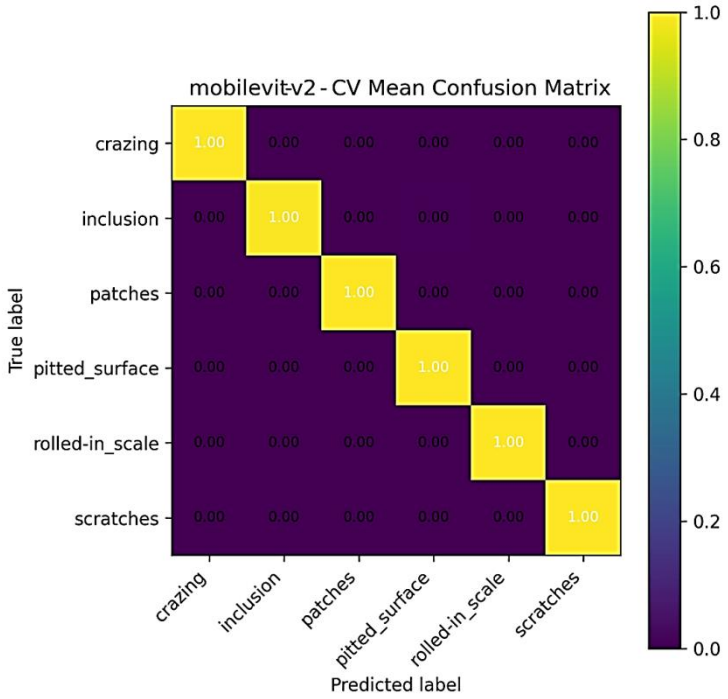


Fig. 10.Average normalized confusion matrix obtained with 5-fold cross-validation.

Examining the confusion matrix results in Figures 9 and 10 reveals consistency with the low-variance performance values obtained in the 5-Fold Cross-Validation process. The high accuracy rates achieved in the test set indicate that the models do not exhibit an overfitting tendency towards specific classes, but rather demonstrate balanced and consistent learning across all defect types. The prominent diagonal structure observed in the confusion matrices suggests that the models do not exhibit a biased tendency towards any particular defect class.

4.4. Explainability (Grad-CAM)

The Grad-CAM method was used to visually examine the decision-making processes of the models. Grad-CAM visualizations clearly show that the models focus directly on the areas where the defect is located, rather than on background textures, when making classifications.As seen in Figures 11 and 12, for each defect class, the models' attention is focused on class-specific symptoms such as crack structures, surface pits, inclusion regions, and scratch lines. This indicates that the high accuracy rates obtained are not random or based on rote memorization, but rather that the models have learned physically meaningful and distinctive surface features. Therefore, Grad-CAM analysis provides significant evidence of explainability supporting the model's generalization ability.In this study, Grad-CAM visualizations were used not to compare performance between models, but to demonstrate the semantic accuracy of the learned representations and the physical consistency of the decision-making process. Therefore, Grad-CAM results are not presented for all models; instead, the MobileViT-V2 and EfficientNetV2-S models, which best represent the study in terms of overall performance and generalization ability, were selected.

Grad-CAM analyses were performed using a completely independent hold-out test set, not a 5-fold cross-validation process. Thus, the visualizations reflect the model's decision-making mechanism on samples it has never seen before. Two samples were selected for each defect class, creating a single grid figure encompassing all classes. This approach allows for a simpler, more understandable, and focused presentation of the explainability analysis. Particularly for defect types such as cracks, scratches, and surface deformations, the high overlap between activation maps and defect boundaries demonstrates that the model bases its classification decisions on physically significant features. Examination of the results supports the idea that the high accuracy values are not based on randomness or data leakage, but rather that the model has truly learned to distinguish defect regions. In this context, Grad-CAM has been used as a qualitative validation tool to support quantitative performance metrics and, consequently, as a decisive criterion in model comparisons, which is the main objective of this study.

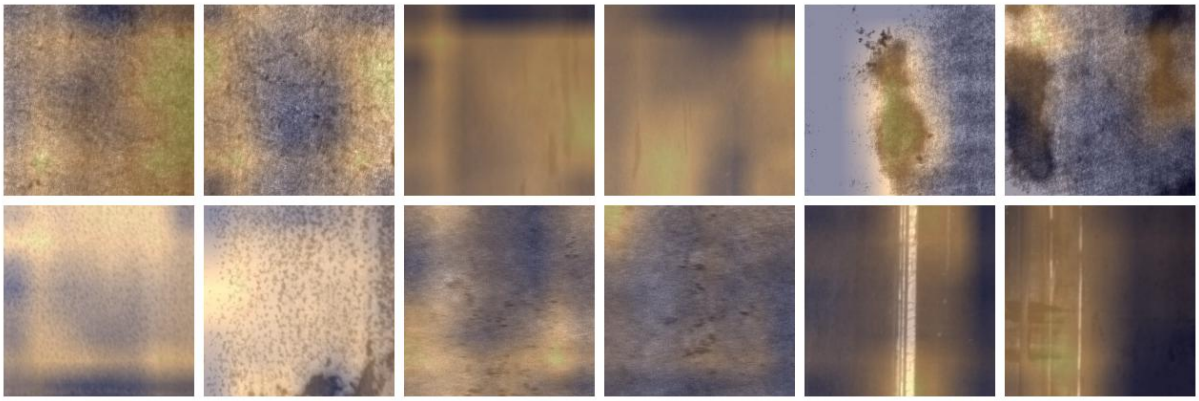


Fig. 11.Grad-CAM visualizations created by selecting two examples from each class in the MobileViT-V2 model testset

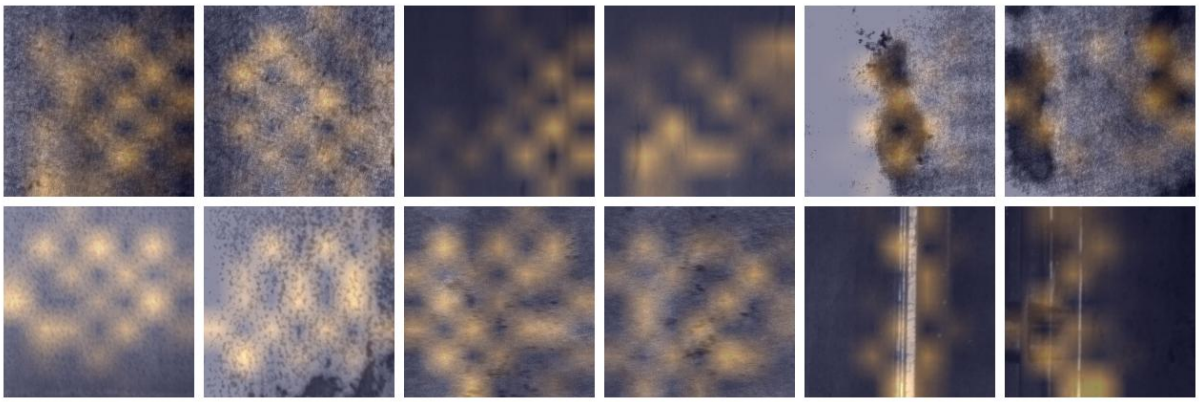


Fig. 12. Grad-CAM visualizations created by selecting two examples from each class in the EfficientNetV2-S model testset

V. Conclusion

This study compares and thoroughly evaluates deep learning architectures that have demonstrated outstanding performance in the literature regarding the problem of classifying industrial surface defects. The experimental results obtained were rigorously examined using both decoupled test data and a 5-fold cross-validation method. Performance evaluation was not limited solely to accuracy metrics; the generalizability and explainability aspects of the models were also considered.

EfficientNetV2-S, Swin Transformer-Tiny, ViT-Small, and MobileViT-V2 architectures all demonstrated the best performance, achieving 100% accuracy and a 100% macro-F1 score on the test data. Furthermore, a detailed examination of the 5-fold cross-validation results reveals that the MobileViT-V2 model produced both more consistent results and exhibited lower variability. This model's ability to achieve higher success using fewer parameters demonstrates that larger, more complex models are not always necessary. Therefore, this can be considered a significant finding in terms of the balance between performance and computational cost.

Future studies aim to test the proposed method on different industrial surface datasets and to examine its generalizability in more challenging and noisy real-world conditions. Furthermore, the contribution of combining multi-scale features, self-supervised pre-training approaches, and data enhancement techniques to model performance can be investigated in more detail. For real-time applications, a significant advantage can be gained by reducing system hardware requirements through the use of model compression and acceleration techniques.

References

[1] Xie, X. (2008). A review of recent advances in surface defect detection using texture analysis techniques. ELCVIA: electronic letters on computer vision and image analysis, 1-22.

[2] Yun, J. P., Shin, W. C., Koo, G., Kim, M. S., Lee, C., & Lee, S. J. (2020). Automated defect inspection system for metal surfaces based on deep learning and data augmentation. Journal of Manufacturing Systems, 55, 317-324.

[3] Fu, G., Zhang, Z., Le, W., Li, J., Zhu, Q., Niu, F., ... & Shen, Y. (2023). A multi-scale pooling convolutional neural network for accurate steel surface defects classification. Frontiers in Neurorobotics, 17, 1096083.
[4] Song, K.,& Yan, Y. (2013). A noise robust method based on completed local binary patterns for hot-rolled steel strip surface defects. Applied Surface Science, 285, 858-864.
[5] Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., ... & Houlsby, N. (2020). An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929..
[6] Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., ... & Guo, B. (2021). Swin transformer: Hierarchical vision transformer using shifted windows. In Proceedings of the IEEE/CVF international conference on computer vision (pp. 10012-10022).
[7] Zhu, W., Zhang, H., Zhang, C., Zhu, X., Guan, Z., & Jia, J. (2023). Surface defect detection and classification of steel using an efficient Swin Transformer. Advanced Engineering Informatics, 57, 102061.
[8] Tang, B., Song, Z. K., Sun, W., & Wang, X. D. (2023). An end-to-end steel surface defect detection approach via Swin transformer. IET Image Processing, 17(5), 1334-1345.
[9] Assad, S., Isa, N. A. M., & Saleh, S. A. M. (2026). Hybrid CNN-Transformer Models for Industrial Defect Detection: A Systematic Review. Results in Engineering, 109457.
[10] Song, H. (2025). RSTD-YOLOv7: a steel surface defect detection based on improved YOLOv7. Scientific Reports, 15(1), 19649.
[11] Zhou, S., Cai, Y., Zhang, Z., & Yin, J. (2025). MESC-DETR: An improved RT-DETR algorithm for steel surface defect detection. Electronics, 14(11), 2232.
[12] Li, Z., Wei, X., Hassaballah, M., Li, Y., & Jiang, X. (2024). A deep learning model for steel surface defect detection. Complex & Intelligent Systems, 10(1), 885-897.
[13] He, Y., Song, K., Meng, Q., & Yan, Y. (2019). An end-to-end steel surface defect detection approach via fusing multiple hierarchical features. IEEE transactions on instrumentation and measurement, 69(4), 1493-1504.
[14] Dixit, K. (2020). NEU Surface Defect Database. Kaggle. https://www.kaggle.com/datasets/kaustubhdikshit/neu-surface-defect-database
[15] Liu, Z., Mao, H., Wu, C. Y., Feichtenhofer, C., Darrell, T., & Xie, S. (2022). A convnet for the 2020s. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (pp. 11976-11986).
[16] Tan, M.,& Le, Q. (2021). Efficientnetv2: Smaller models and faster training. In International conference on machine learning (pp. 10096-10106). PMLR.
[17] Dai, Z., Liu, H., Le, Q. V., & Tan, M. (2021). Coatnet: Marrying convolution and attention for all data sizes. Advances in neural information processing systems, 34, 3965-3977.
[18] Mehta, S.,& Rastegari, M. (2021). Mobilevit: light-weight, general-purpose, and mobile-friendly vision transformer. arXiv preprint arXiv:2110.02178.