Proceedings of 8th International Conference on Mechanical Engineering and Renewable Energy 2025
10-12 December, 2025, Chattogram, BANGLADESH
SciEn Conference Series: Engineering Vol. 4, 2026, pp 253-258 https://doi.org/10.38032/scse.2026.4.120

Defect Detection on Metal Surface Using Deep Learning Algorithms Based on Convolutional Neural Network

Prottoy Dev[1,], Aninda Chowdhury[2] and Md Abdul Ahad Abir[1]*

[1]Department of Mechanical Engineering, Faculty of Mechanical & Manufacturing Engineering, Chittagong University of Engineering & Technology, Chattogram-4349, Bangladesh
[2]Department of Leather Engineering, Faculty of Mechanical Engineering, Khulna University of Engineering & Technology, Khulna-9203, Bangladesh

ABSTRACT

This project aims to develop a deep learning model utilizing Convolutional Neural Networks (CNNs) to detect and categorize surface defects in metal photos. Traditional manual inspection is typically slow, uneven, and full of mistakes, with a research gap in automated solutions. The objective is to provide an effective and dependable solution for fault identification, crucial in manufacturing and quality control procedures. A proprietary CNN will be trained on a large dataset of labeled pictures, facilitating the extraction of critical characteristics via convolutional, pooling, and fully connected layers. The project will examine the usage of pre-trained models such as VGG16 and ResNet50. These models will be evaluated singly and in combination using an ensemble method to boost classification accuracy and resilience. The models' performances will be measured based on evolution metrics such as accuracy, loss, and their ability to categorize photos into one of the defect categories. It is predicted that the ensemble strategy would give greater performance compared to individual models, illustrating the usefulness of deep learning-based methods for automated fault identification. The method will give a scalable deep learning framework for defect detection, enabling real time integration in manufacturing industries.

Keywords: Convolutional Neural Networks, Surface Defects, VGG16, ResNet50, Deep Learning.

1. Introduction

Finding faults on metal surfaces is a vital aspect of quality assurance in the industrial sector. This process has typically been based on manual inspection, in which professional inspectors visually check steel surfaces for faults. Although hand inspection is labor-intensive, time-consuming, and prone to human mistakes, it can be effective in some cases. Inspector tiredness might result in uneven and untrustworthy conclusions. Since a large percentage of the completed goods are characterized by surface imperfections, production technology needs to be improved [1]. With the rapid growth of artificial intelligence, it has become viable to employ machine vision instead of people to tackle such challenges. Machine vision-based methods for finding surface defects have been more popular in recent years because they fix some of the problems with manual detection, such as low accuracy, poor real-time performance, subjectivity, and high labor intensity [2]. Even when these procedures were slightly better than human inspection, they frequently couldn't manage the intricacy and unpredictability of genuine faults on steel surfaces. Although automated inspection systems have demonstrated promise, many of the prior approaches, such as traditional machine learning techniques and basic image processing have not been able to manage the complexity of surface faults in steel or metals. Improvements in image processing technology, in the field of machine vision, have led to the creation of automated flaw detection algorithms [1]. Defect detection has experienced a major transformation with the introduction of deep learning. Deep learning models may enhance detection consistency and accuracy by automating the inspection process and minimizing reliance on human inspectors.

Fig.1: Real time defect images of metal surfaces [3].

Lower manufacturing costs, less waste, and greater product quality ensue from this. In this research, an automated flaw identification system is created for metallic surfaces employing deep learning models, such as a bespoke CNN, VGG16, and ResNet50. Specifically intended to recognize patterns and characteristics connected to different forms of problems, the custom CNN can detect weaknesses on metal surfaces. Well-known models VGG16 and ResNet50, which were initially built for image classification, have been updated to match the characteristics of metal surface defect detection. Since 2015, residual neural networks have shown great achievements in image processing [5] ResNet50 displays great generalization performance with lower error rates on picture recognition tests and is, thus, a helpful tool for categorization [6]. While ResNet50 employs skip connections to address gradient vanishing concerns and increase model performance and stability, VGG16 focuses greater emphasis on depth in its design. The main objective is designing a dependable system that leverages advanced machine learning models to precisely identify and categorize steel surface faults. To identify steel surface faults, adapt and fine-tune deep learning algorithms and models including CNN, VGG16, and ResNet50 models, to boost their accuracy, efficiency and other metrics. Then to build an ensemble model using other models for better accuracy. The fundamental motive for putting these approaches into reality is to make steel companies more efficient in the international market.

2. Materials And Method

This methodology part includes a full and rigorous process for designing, training, and analyzing the CNN model for precise metal surface defect detection. This strategy is meant to assure a thorough understanding and execution of each process, from data collecting to deployment and performance monitoring. Initially, it entails the collection and processing of the data, which acts as the foundation for training and evaluating the model. Following this, the data pretreatment procedures, including normalization, scaling, and augmentation, are presented to guarantee the data is in optimum condition for model training.

2.1 Data Collection

The NEU Metal Surface Defects Data comprises pictures exhibiting six frequent kinds of defects detected as shown in Fig.01 hot-rolled steel strips, namely crazing, inclusion, patches, pitted surface, rolled-in scale, and scratches. This dataset gives a robust base for the development and testing of 18 defect detection models. The photos in this dataset have a constant size of 200x200 pixels, which makes them acceptable for input into convolutional neural networks (CNNs). To increase the strength and flexibility of the defect detection models, we will acquire additional photographs from diverse industries.

2.1.1 Data Preparation

Normalize and augment data with techniques like rotation, zoom, and flipping. Splitting data into training (70%), validation (15%), and testing (15%).

2.2 Training Models

The training procedure will begin by separating the data set into training and validation sets. The models will be made using the training set, and the validation set will check how well they work during the training phase. This measure will mitigate overfitting by ensuring that the models generalize effectively to novel data. Each model—custom CNN, VGG16-based, and ResNet50-based—will adhere to its own architecture throughout the training process. The bespoke CNN will depend on hand crafted convolutional and pooling layers specifically developed for the dataset. Conversely, the VGG16 and ResNet50 models will utilize pre-trained layers from ImageNet, with VGG16 emphasizing fast transfer learning and ResNet50 using deeper residual connections to capture complex characteristics in the defect pictures.

2.2.1 Custom CNN Architecture

The proprietary CNN architecture has convolutional layers that pull-out features, max-pooling layers that cut down on the number of dimensions, and fully connected layers that sort things into groups. Pooling makes things less complicated, whereas convolutional layers find edges, forms, and patterns. Lastly, fully linked layers use characteristics to make accurate predictions about outputs, which makes finding and classifying defects faster.

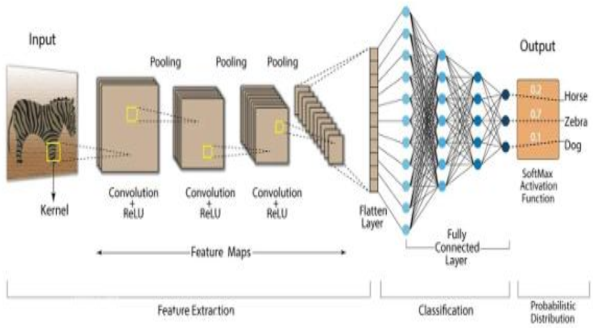

Fig. 2 Basic architecture of a CNN [7]

2.2.2 VGG16 Based Model

The VGG16-based model will apply transfer learning to classify surface flaws in metal photos. VGG16 is a pre-trained convolutional neural network noted for its deep, sequential design, first trained on the ImageNet dataset. By recycling its learnt features, the model will efficiently adapt to the unique dataset, conserving computing resources while keeping powerful feature extraction capabilities.

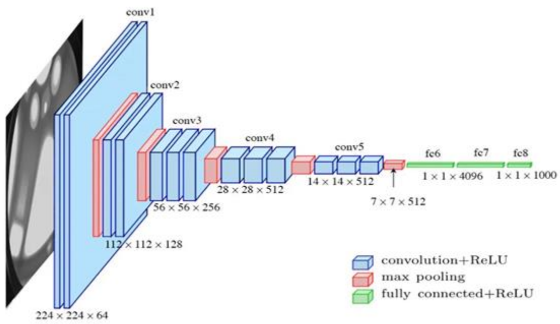

Fig.3 VGG16 Architecture [8].

2.2.3 ResNet50 Based Model

The ResNet50-based model will likewise incorporate transfer learning but with a deeper architecture and unique residual connections. ResNet50 is a state-of-the-art deep learning model pre-trained on ImageNet, characterized by its ability to alleviate difficulties like disappearing gradients. Its architecture makes it particularly good at collecting complicated patterns and hierarchies in visual data.

2.4 Ensemble Strategy

Ensemble learning is a strategy that amalgamates the forecasts of several models to create a singular, more exact forecast.

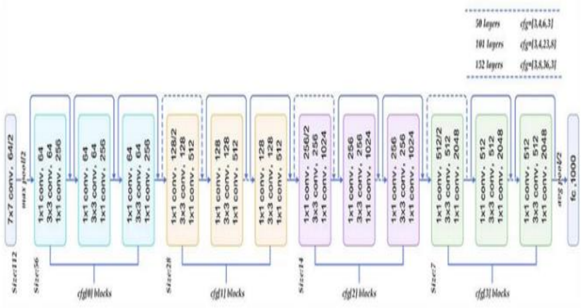

Fig.4 Residual Network 50 Architecture[9].

Ensemble techniques give numerous major benefits, including lower variance, greater generalization, and improved control of overfitting. This project employs a mix of three models: a proprietary Convolutional Neural Network (CNN), VGG16, and ResNet50 After training the individual models, the ensemble approach integrates their predictions. In this research, a basic yet successful strategy is employed: the weighted average of the projections. The tailored Convolutional Neural Network (CNN), with its uniquely constructed structure, excels in properly identifying and capturing distinguishing properties that are important to metal faults. The VGG16 and ResNet50 models, with their deep architectures and pretrained weights, give strong feature extraction capabilities that boost the overall performance of the ensemble. In summary, the ensemble strategy in this work combines the predictions of a custom CNN, VGG16, and ResNet50 using a weighted average approach. This technique maximizes the accuracy and robustness of fault identification, utilizing the characteristics of each individual model and providing dependable performance in real-world applications.

2.5 Evaluation Metrics

Evaluating the performance of the models is crucial to determining their efficacy and dependability. Accuracy is a measure that quantifies the ratio of accurately recognized photographs to the total number of images.

$$\text{Accuracy} = \frac{\text{Number of Correct Predictions}}{\text{Total Number of Predictions}} \quad (1)$$

Precision is a statistic that calculates the ratio of accurately anticipated positive cases to the total number of expected positive instances. It assesses the accuracy of positive predictions by assessing the frequency of occurrences that were anticipated as positive and are genuinely positive.

$$\text{Precision} = \frac{\text{True Positve}}{\text{True Positve+False Positve}} \quad (2)$$

Recall quantifies the proportion of correct positive forecasts out of the total genuine positives. It shows the model’s capacity to capture all relevant occurrences in the dataset.

$$\text{Recall} = \frac{\text{True Positives}}{\text{True Psoitives+False Negatives}} \quad (3)$$

The F1-value is the harmonic mean of accuracy and recall, delivering a single value that balances both objectives. It is particularly effective when there is an unbalanced class distribution or when both false positives and false negatives incur large costs.

$$\text{F1 Score} = 2\times\frac{\text{Precision}\times\text{Recall}}{\text{Precision+Recall}} \quad (4)$$

The confusion matrix shows the number of true positives, true negatives, false positives, and false negatives, which gives a complete picture of how well the model works.

3.Results And Discussion

3.1 Data Collection

This work uses the NEU Metal Surface Defect Database, a publicly accessible benchmark dataset established by the Institute of Automation, Chinese Academy of Sciences, as the major dataset for training and analyzing the defect identification algorithms. This dataset is commonly employed for surface defect recognition tasks and contains high-resolution industrial images acquired under controlled lighting and positioning conditions to eliminate background noise and inconsistencies. Each image in the collection is 200 × 200 pixels and is in grayscale format, exhibiting diverse fault textures within a constant monochrome context. This design choice is very important since using grayscale images makes deep learning models less complicated and gets rid of noise linked to color, which is not needed for jobs that find defects based on texture. This study's models were only trained on grayscale data.

In computer vision, the conversion of a color image (RGB) to grayscale requires converting the three-color channels (Red, Green, and Blue) into a solitary intensity channel. This is performed by finding a weighted sum of the RGB values with the formula.

$$\text{Grayscale} = 0.2989\times R+0.5870\times G+0.1140\times B \quad (5)$$

The datasets look like as shown:

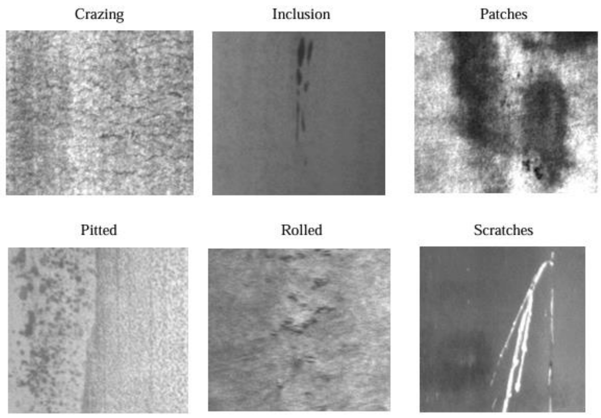

Fig.5: Images from NEU Metal Surface Defect Dataset

3.2 Data Preparation

There were six files in the NEU surface defect dataset at first, and each file had a different form of defect. The dataset had 1,800 grayscale pictures, which were split into three groups: training (1,464), validation (264), and test (72). Each subset maintained consistent representation across all six fault categories to ensure fair learning and evaluation. We have used Keras's Image Data Generator to preprocess and add to the data. This made it possible to load, add to, and batch the data in real time during training. For the training set, the following preprocessing operations are firstly pixel values were normalized from [0, 255] to [0, 1] using rescale=1./255, random rotation within ±20 degrees, width and height shift up to 10% of the image dimensions, horizontal flipping then Shuffling of images for each epoch.

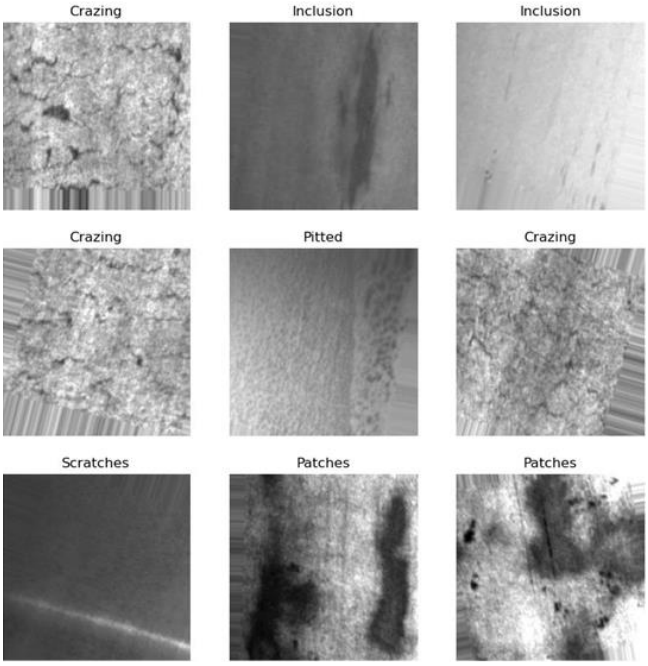

Fig.6: Preprocessed Images of Datasets

Each subset was loaded using the flow_from_directory function with a batch size of 32, allowing efficient mini-batch training and memory management

3.3 Model Training and Setup

Custom CNN, VGG16, and ResNet50 were trained in the identical settings for fair comparison. All the models employed the Adam optimizer with a learning rate of 0.001 And categorical cross-entropy loss. To avoid stagnation during training, a ReduceLROnPlateau callback was employed across all models. This callback assessed either the validation loss (for VGG16 and ResNet50) or validation accuracy (for the Custom CNN) and dropped the learning rate by a factor of 0.1 if the metric did not improve after a few epochs. This dynamic scheduling helps the model converge more successfully when learning slows down.: All models were trained for 30 epochs with a batch size of 32, which provides a compromise between training stability and computational efficiency. An epoch refers to one complete journey throughout the whole training dataset. In each epoch, the model processes all training photos once, learns from them, and updates its internal parameters depending on the loss and gradients. Repeating this strategy throughout multiple epochs helps the model produce increasingly accurate representations over time.

.3.3.1 Baseline CNN Model

4 convoluted layers used. Maxpooling2D was used between each layer. ‘Relu’ and ‘SoftMax’ were used as activation functions. Batch normalization was used. 20% of neural networks dropped out. Categorical cross entropy was used as loss function. Adam optimizer was used with base learning rate of 0.001. It was reduced by using a scheduler.

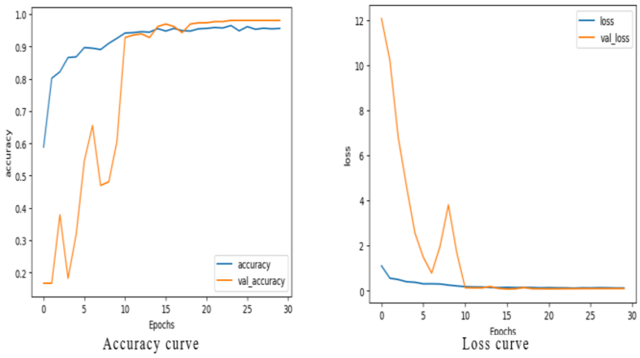

Fig.7: Accuracy and Loss curves of Baseline CNN model

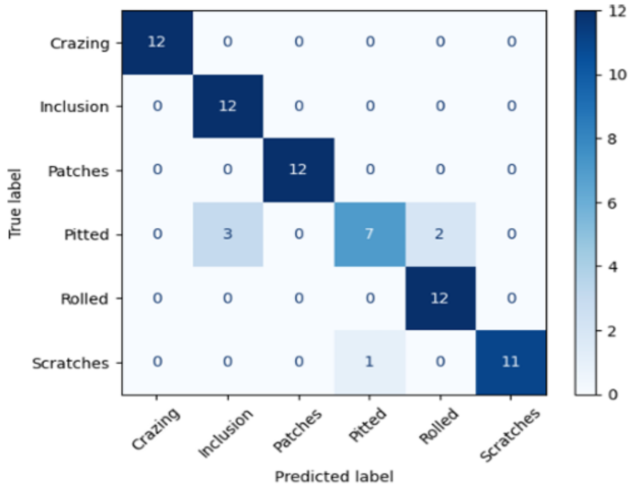

Fig.8: Confusion matrix for the Baseline CNN model.

3.3.2 VGG16 Model

It has 16 convoluted layers. ImageNet weights were used. ‘Relu’ and ‘Softmax’ used as activation function. 30% of neural networks dropped out and Categorical crossentropy was used as loss function. Adam optimizer was used with base learning rate of 0.001. It was reduced by using a scheduler.

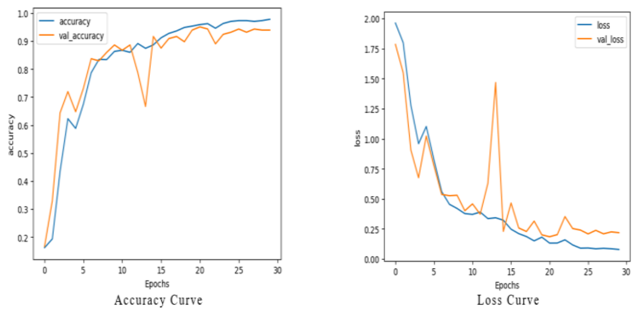

Fig.9: Accuracy and Loss curves of the VGG16 model

Result summary for VGG16 model:91.67% accuracy on the test data. Six misclassifications on the test data. Minor misclassification on Pitted type defects.

3.3.3 ResNet50 Model

50 convoluted layers. ImageNet weights were used. ‘Relu’ and ‘Softmax’ used as activation function. 20% of neural networks dropped out. Categorical cross entropy was used as loss function. Adam optimizer was used with base learning rate of 0.001. It was reduced by using a scheduler.

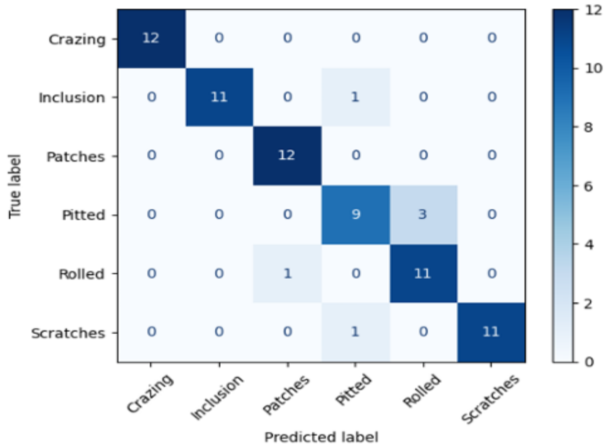

Fig.10: Confusion matrix for the VGG16 model.

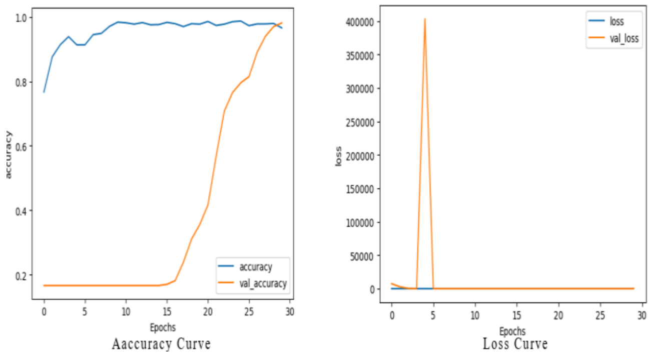

Fig.11: Accuracy and Loss curves of the ResNet50 model

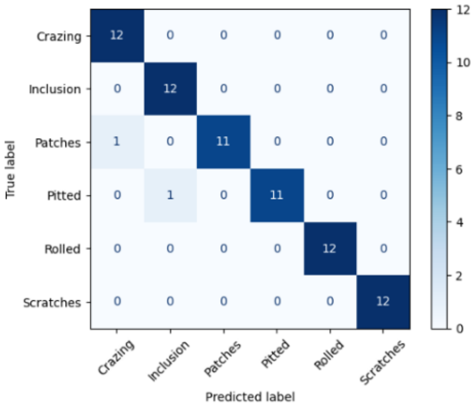

Fig.12: Confusion matrix for the ResNet50 model.

Result summary for ResnNet50 model are 97.22% accuracy on the test data's. Two misclassifications on the test data.

3.3.4 Stacked Ensemble Model

To increase classification performance by utilizing the properties of several models, a stacked ensemble model was built. This ensemble architecture integrates the prediction probabilities of three individually trained base models—Baseline CNN, VGG16, and ResNet50. Each base model first provided prediction probabilities on the validation set, which were horizontally stacked to give the input features for a meta-classifier. A logistic regression model was then trained on these stacked predictions using the true validation labels. During the test phase, the same stacking technique was applied using test set predictions, and the trained logistic regression meta-model yielded the final predicted labels.

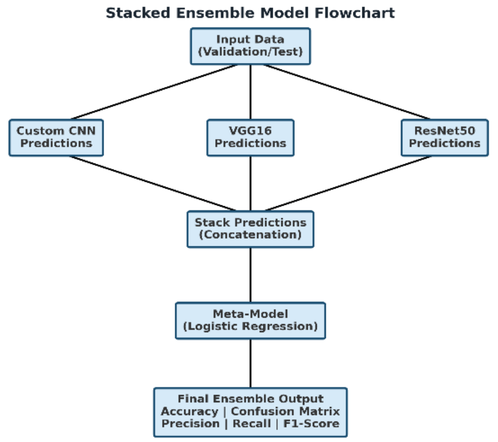

Fig.13: Flowchart of Ensemble Training Model

3.4 Ensemble Result & Comparison

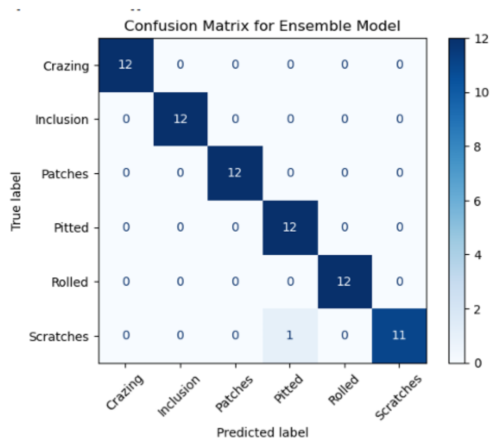

Fig.14: Confusion matrix for the Ensemble model

The ensemble model demonstrated a superior generalization capacity, obtaining an overall test accuracy of 98.61% which greatly outperformed all individual models. The performance was then assessed using a confusion matrix, which clearly indicated near-perfect classification across all fault classes. Only one misclassification on the Scratch type defect. To study and compare the performance of each model more extensively, additional assessment metrics—Precision, Recall, F1-Score, and Accuracy—were applied alongside the provided bar chart. These metrics offer a fuller perspective into each model's potential to recognize and classify surface faults across all categories, which is especially essential in multi class classification jobs.

All individual models performed well, with both the CNN and VGG16 models getting 91.67% accuracy and an F1-score around 0.91–0.92. The ResNet50 model shone out among the basic models, getting a superior accuracy of 97.22% and F1-score of 0.97.

This shows that the ensemble approach successfully exploits the peculiarities of each unique design to produce a more reliable and accurate classification system for surface fault diagnosis.

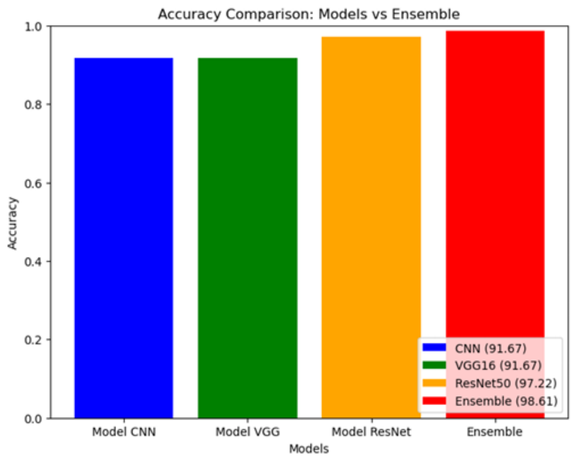

Fig.15: Comparison of the models

Table 1: Evaluation Metrics Comparison of All Model

Model	Accuracy	Precision	Recall	F1-Score
Baseline CNN	91.67%	0.92	0.92	0.91
VGG16	91.67%	0.92	0.92	0.92
ResNet50	97.22%	0.97	0.97	0.97
Ensemble	98.61%	0.99	0.99	0.99

3.5 Ablation Study on Custom CNN

The baseline CNN with four convolutional layers, batch normalization and dropout the stable and generalized performance.

Table 2: Results of Ablation Study on the Custom CNN

Change Made	Test Accuracy (%)	Observation
Original 4-layer model	91.67	Stable and well-generalized
One convolutional layer removed	88.88	Lower feature learning capacity
One additional layer added	90.27	Slight overfitting, unstable validation
Batch normalization removed	86.11	Unstable learning, poor generalization
Dropout removed	89.44	Mild overfitting, higher validation variance

4. Conclusion

This study's results proved the usefulness of deep learning in detecting surface flaws. The baseline CNN model reached an accuracy of 92%, whereas VGG16 and ResNet50 acquired accuracy of 93.56% and 97%, respectively. The stacked ensemble model beat all individual models, obtaining an accuracy of 98.6%, therefore proving the usefulness of merging numerous classifiers to boost robustness and generalization. Notably, the ensemble model exhibited almost faultless performance with only a single misclassification, showing its dependability for practical use. Future work may focus on real-time deployment, expanding to larger or more complex datasets, and integrating object detection models like YOLO or vision transformers.

References

[1] I. Konovalenko, P. Maruschak, V. Brevus, and O. Prentkovskis, “Recognition of scratches and abrasions on metal surfaces using a classifier based on a convolutional neural network,” *Metals (Basel)*, vol. 11, no. 4, Apr. 2021,

[2] X. Tao, D. Zhang, W. Ma, X. Liu, and De Xu, “Automatic metallic surface defect detection and recognition with convolutional neural networks,” *Applied Sciences (Switzerland)*, vol. 8, no. 9, Sep. 2018,

[3] S. K. Dhua, “Metallurgical Analyses of Surface Defects in Cold-Rolled Steel Sheets,” *Journal of Failure Analysis and Prevention*, vol. 19, no. 4, pp. 1023–1033, Aug. 2019, n.

[4] Y. Xu, K. Zhang, and L. Wang, “Metal surface defect detection using modified yolo,” *Algorithms*, vol. 14, no. 9, Sep. 2021.

[5] Xu Ke and Yang Chaolin, “On-line defect detection algorithms for surface inspection of hot rolled strips,” in *2010 International Conference on Mechanic Automation and Control Engineering*, IEEE, Jun. 2010, pp. 2350–2353.

[6] I. Konovalenko, P. Maruschak, J. Brezinová, J. Viňáš, and J. Brezina, “Steel surface defect classification using deep residual neural network,” *Metals (Basel)*, vol. 10, no. 6, pp. 1–15, Jun. 2020,

[7] Rismayani, A. A. Ilham, A. Achmad, and M. R. Y. Rachman, “Convolutional Neural Network with Feature Extraction to Improve the Classification Accuracy of Multi-Class Facial Skin Disorders,” *International Journal of Online and Biomedical Engineering*, vol. 21, no. 3, pp. 4–19, Mar. 2025,

[8] Khuyen Le, “An overview of VGG16 and NiN models,” Medium. Accessed: Dec. 28, 2024. [Online].Available:https://lekhuyen.medium.com/an-overview-of-vgg16-and nin-models-96e4bf398484.

[9] Aditi Rastogi, “ResNet50. ResNet-50 is a convolutional neural network,” Medium. Accessed: Dec. 28, 2024.