Comparative Analysis of Fine-tuning Multiple Pre-Trained Convolutional Neural Network (CNN) Models for *Oryza Sativa* Disease Detection

Roky Das
B.Sc. Engineering,
Dept. of CSE
University of Chittagong,
Chattogram, Bangladesh

Iqbal Ahmed, PhD
Professor,
Dept. of CSE,
University of Chittagong,
Chattogram, Bangladesh

ABSTRACT
Rice is the staple food in Bangladesh. Every year many crops are lost because of the disease of the rice plant. Our farmers are facing a great loss every year as well as rice productivity is declining. Some diseased and healthy leaf images of rice plants have been collected. The images have been classified into four classes. The classes are Brown Spot, Hispa, Leaf Blast diseased leaves, and Healthy leaves, usually known as Oryza Saiva diseases of rice. Deep learning is very common and popular to analyze and make predictions from these kind of image data. In this research, the CNN (Convolutional Neural Network) models have been used to predict the classes. Among various CNN models, VGG16, MobileNet, ResNet50, and DenseNet121 pre-trained models have been applied. Among those pre-trained models, the highest accuracy 95.3% has been found from the VGG16 model. In addition, an Android mobile app has been developed using the highest performant VGG16 trained model. Using this mobile app, farmers will be able to upload a photo and predict the disease of that rice plant or not in a convenient way.

Keywords
CNN, Deep Learning, Convolutional Neural Network, *Oryza Sativa*, Rice plant disease

1. INTRODUCTION
Rice is one of the most important world's three main crops. Human consumption accounts for 78% of the total production of rice [1,2,12]. Rice is the staple food of more than half of the global population – More than 3.5 billion people depend on rice for more than 20% of their daily calories. Asia accounts for 90% of global rice devouring, and total rice demand is emerging there. However, outside Asia, where rice is not a staple yet, per capita consumption continues to grow.

Many studies have already been conducted to recognize rice diseases [1,3,5,9]. Most of these studies have used machine learning algorithms like support vector machines and genetic algorithms where features are extracted and used to classify different diseases. The performance of Convolutional Neural Networks (CNN) in image classification has recently been very promising, leading researchers to consider using this technique to identify rice diseases as well [3,8,12,13]. This study has applied a Convolutional Neural Network for classifying rice diseases from images.

Convolutional Neural Networks can extract the features themselves which ensures no important features get missed during the feature extraction process. This is the advantage of this method that it can save the cost of extracting the features which is computationally very expensive.

2. RELATED RESEARCHES
A technique for identifying fast rice illness based on the fusion of FCM-KM and Faster R-CNN is presented in [1] to solve a number of issues with the pictures of the disease, including noise, fuzzy image edges, significant background interference, and poor detection accuracy. First, a faster two-dimensional Otsu threshold segmentation algorithm (Faster 2D-Otsu) is used to reduce interference from a complex background with the detection of the target blade in the image. The method also uses a weighted multilevel median filter in combination with a two-dimensional filtering mask to reduce noise. To ascertain the various sizes of the Faster R-CNN target frame, FCM-KM analysis is carried out in conjunction with the R-CNN algorithm for the detection of rice illnesses.

The most significant crop that has a significant impact on India's economy is rice [2]. Four primary diseases have been identified as having a significant impact in India's rice plant: rice leaf blast, bacterial blight, sheath blight, and brown spot [1]. The goal of this study is to enhance an enhanced system that combines machine learning and image processing [2]. As a result, a farmer may quickly and easily identify the illness and take preventative measures.

In this research [3], a convolutional neural network (CNN) was used to analyze photos in order to detect and classify illnesses. Blast, bacterial leaf blight, brown spot, narrow brown spot, bacterial leaf streak, and rice ragged stunt viral disease are six of the most common illnesses that affect rice. The detection abilities of four well-known pre-trained models—Faster R-CNN, RetinaNet, YOLOv3, and Mask RCNN—are investigated in this work.

According to [4], address the aforementioned issues, a faster region-based convolutional neural network (Faster R-CNN) was used in the current study [4] to identify rice leaf illnesses in real time.

In [5], deep learning-based methods for identifying diseases and pests from photos of rice plants have been created. These methods were inspired by the success of CNNs in image classification. This work makes two contributions: (i) Modern large-scale architectures like VGG16 and InceptionV3 have been used and optimized for identifying illnesses and pests that affect rice. The usefulness of these models with actual datasets is shown by experimental findings. (ii) A two-stage compact CNN design has been presented and contrasted with cutting-edge memory-efficient CNN architectures like MobileNet, NasNet Mobile, and SqueezeNet since big-scale architectures are not appropriate for mobile devices.

By training convolutional neural network (CNN) models using segmented picture data, this research [6] looks at a possible fix for this issue. When evaluated on independent data previously unknown to the models, even with 10 illness classifications, the S-CNN model trained using segmented pictures more than doubles in performance to 98.6% accuracy as compared to the F-CNN model trained using whole images.

Moreover, the goal of the article [7] is to apply CNN and artificial intelligence to identify illnesses that affect rice crops. The diseases that a paddy crop encounters are stored in the database, which is a raspberry pi, and when the farmer takes a picture of the crop, the Pi analyzes it and compares it to the database images using artificial intelligence and convolutional neural network concepts to determine whether the crop is affected by a specific disease or not, at which point it alerts the farmer to the disease.

A CNN model has been used to increase maize leaf disease detection accuracy in the study [8]. The four different types of corn leaf pictures are trained and tested using the upgraded CNN model, which is made possible by adding rectified linear unit activation functions, the Adam optimizer, modifying the parameters, pooling operations, and fewer classifiers.

Farmers would find it hard to regularly assess the huge farmlands. Even if this is feasible, it would be an expensive process that would raise the cost of rice for consumers. A solution to this issue may be provided by machine learning algorithms adapted to drone technology and the Internet of Things (IoT) [9]. In this study, a method is represented for the precise detection and classification of rice leaf disease based on Deep Convolutional Neural Network (DCNN) transfer learning.

With their little understanding, farmers find it very challenging to manually detect these illnesses effectively. Convolutional Neural Network (CNN) models used in automatic image recognition systems may be particularly helpful in solving such issues, according to recent advancements in deep learning [10]. The dataset gathered from rice fields and the internet is used to train and test the suggested CNN architecture, which is based on VGG-16. The suggested model's accuracy is 92.46%.

A strong and all-encompassing method has been put forward, which, by employing the foundations of current CNN models, can identify illnesses in a variety of crops. ResNet-152 and Inception-v3 variations have been suggested for the early identification of illnesses in important crops like rice and maize [11]. The suggested technique, which uses variations of InceptionV3 and ResNet152, has, for corn crop, achieved accuracy of 97.81% and 97.48%, respectively [11]. The photos of rice illnesses are divided into main and minor subgroups to better comprehend the variety of diseases. For the main and minor illness subgroups, the suggested ResNet152 variation has accuracy of 99.10% and 82.20%, respectively. The suggested technique has shown resilience in disease identification.

This research offers two well-known convolutional neural network (CNN) models, AlexNet and ResNet-50, together with image processing as a method for detecting and preventing plants leaf disease in the agricultural area [14]. To study the signs of a diseased leaf, this approach is first used on Kaggle datasets of tomato and potato leaves. Then, using AlexNet and ResNet-50 models together with image processing, the feature extraction and classification procedure is carried out on dataset pictures to identify leaf illnesses.

3. METHODOLOGIES

The approach used to compare and analyze several pre-trained Convolutional Neural Network (CNN) models for spotting disease in *Oryza Sativa* (rice) plants is described in this part. The research attempts to solve the urgent agricultural dilemma faced by Bangladeshi farmers, where significant crop losses are caused by rice plant diseases each year. To address this problem, a dataset of photos of sick and healthy rice plant leaves was gathered and divided into four classes: diseased leaves with Brown Spot, Hispa, Leaf Blast, and Healthy leaves. The study makes use of pre-trained CNN models from deep learning, especially for image analysis and classification applications. The choices for illness categorization included the well-known VGG16, MobileNet, ResNet50, and DenseNet121 pre-trained CNN models. The models that were selected were customized to the issue of rice disease detection by fine-tuning and transfer learning. The VGG16 model was used to attain the greatest accuracy of 95.3% after the training and assessment phases were completed. The best performing VGG16 model was then used to create an Android mobile application that enables farmers to take and submit photos of rice plants for disease prediction. The above-mentioned overall procedure is shown in next figure 1. The specifics of each part of the process are covered in depth in the following sub sections.

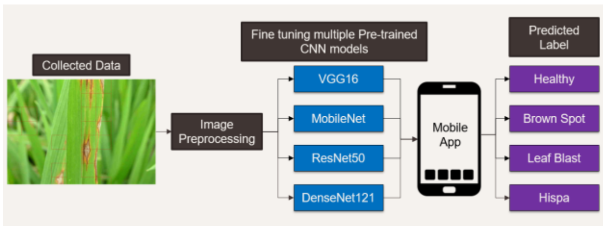

Fig. 1: Overall research methodologies

3.1 Data Collections

In research and analysis, acquiring correct and pertinent information from a variety of sources is known as data collection. Any study's success and validity greatly depend on the accuracy and reliability of the data gathered. Numerous techniques, including surveys, experiments, observations, interviews, and already-existing databases, may be used to gather data. In this research, dataset is collected from the Kaggle website. In following table 1, distribution of various classes are given and total 6061 image data have been collected.

Table 1: Dataset Analysis

Disease	Count
Brown Spot	965
Healthy	1764
Hispa	1594
Leaf Blast	1738
Total	**6061**

Each class's demo image is represented in next figure 2.

Fig. 2: Demo of each class

3.2 Data Preprocessing

Data preprocessing, a crucial stage in the pipeline for data analysis and machine learning, entails converting unstructured raw data into a format that can be used. The primary goal of data preprocessing is to improve the data's quality and relevance so that it is appropriate for subsequent analysis and modeling.

3.3 Model Training

Model training is the process of exposing a machine learning model to a labeled dataset to educate it on how to make precise predictions. The objective is to make it possible for the model to discover patterns, connections, and rules from the data. The CNN (Convolutional Neural Networks) model has been used in this research (figure 3). Nowadays, CNN is outperformed in the field of deep learning. The training dataset was fed to the CNN model. They were fed to some pre-trained CNN models. Then the results were compared, and the best performing model among the models was selected. The used pre-trained models are VGG16, MobileNet, ResNet50, and DenseNet121. The dataset was divided into subgroups for training, validation, and testing. The quantity of the training, validation, and testing dataset was 70%, 15%, and 15% of the whole dataset respectively.

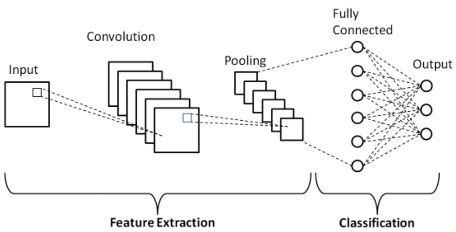

Fig. 3: Convolutional Neural Network.

3.4 Model Evaluation

A trained machine learning model's quality and performance are evaluated via model evaluation. The goal is to evaluate how well the model does at generating predictions based on fresh, unforeseen data. Few criteria have been selected to compare among the four models. These criteria are Validation accuracy, Validation loss, Test accuracy, Confusion matrix, Precision, Recall, and F1-score.

3.5 User Interface Development (Android Mobile App)

User Interface Development is the process of making a trained machine-learning model usable in real-world applications or production settings. Making the model accessible for use in practice entails taking the model that was created and put through testing during the research phase.

The process of making a trained machine-learning model usable in a production context is known as User Interface development. A model's integration into actual applications, systems, or services once it has undergone training and evaluation allows end users to take advantage of its predictive capabilities. User Interface Development entails adapting the trained model to the deployment environment's specifications, improving its performance, and putting in place any infrastructure required to enable its execution. Therefore, the model is guaranteed to be able to handle prediction requests effectively and to provide trustworthy, scalable results.

4. RESULTS AND DISCUSSION

In this research, four pre-trained famous CNN models were trained using some images of rice leaves. Three types of diseased leaves and healthy leaves were classified. Results from every model have been found, and in this section, the results will be discussed. Some common machine-learning criteria have been selected to compare among them. Four models have been selected, - VGG16, MobileNet, ResNet50, and DenseNet121. The accuracy of the VGG16 model is the highest among all the models which is 94.5% and testing accuracy is 95.3%. The validation accuracy and testing accuracy of all models are shown in figure 4. Since VGG16 is the highest performer, an Android Mobile App was built to be used as an interface for our model. Farmers will be able to use this app directly.

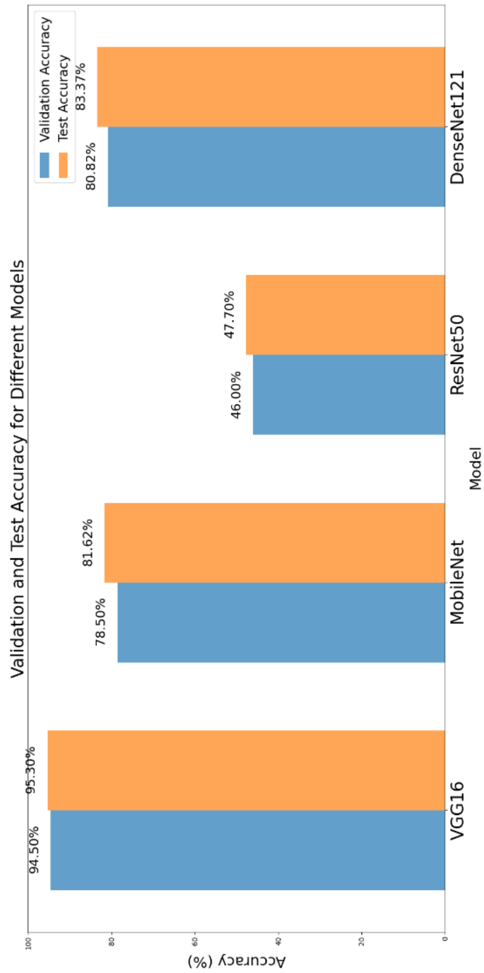

Fig. 4: Validation and Testing Accuracy of all models

4.1 VGG16 CNN Model

It is one of the most used pre-trained CNN models. It was used in our research, and it out performed among all models.

4.1.1 Performance of VGG16 CNN Model

In figure 5, the "Epoch vs Accuracy" graph and in figure 6, the "Epoch vs Validation Loss" graph is shown for VGG16 CNN model. Validation accuracy is increasing according to the increase of epochs and validation loss is decreasing according to the increase of epochs. Therefore, it is evident that VGG16 CNN model is performing well in this research.

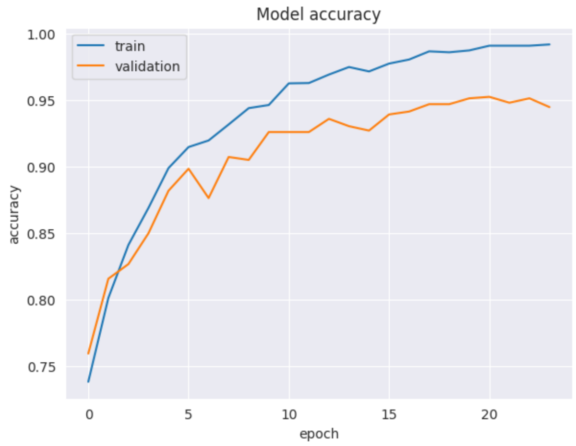

Fig. 5: Epoch vs Accuracy Graph of VGG16

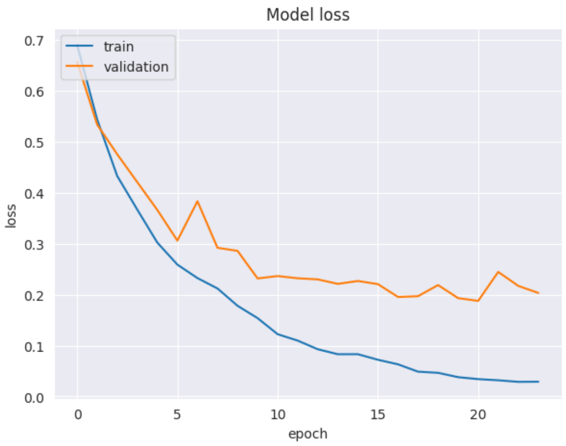

Fig. 6: Epoch vs Validation Loss Graph of VGG16

4.1.2 Confusion Matrix of VGG16 CNN model

In the following figure 7, the confusion matrix of the VGG16 CNN model is represented. In the matrix, for Brown Spot class, all are predicted correctly and for healthy class, only 10 are predicted wrong whereas 255 are predicted well. For Hispa, the right prediction number is 226 and the wrong prediction number is 6. For the Life Blast, the right is 259 and the wrong is 18. Therefore, it can be concluded that the VGG16 CNN model is performing well.

4.1.3 Result of VGG16 CNN model

After evaluating the VGG16 pre-trained CNN model using the testing dataset, the result was found and represented in Table 2. Also, the Accuracy, Macro Average, and Weighted Average are shown in the Table 2.

With a 95.7% overall accuracy rate, the model successfully predicts the class labels for most cases in the dataset. The accuracy values for all classes range from 93.5% to 97.4%, which is a reasonably high figure. This indicates that the model's predictions for a given class are often correct and have a low probability of false positives. The recall values vary from 92.5% to 98.9%, indicating how well the model can detect instances of each class. With a low percentage of false negatives, this shows that the model does a good job of capturing most occurrences of each class. 94.4% to 96.1% is the range of the F1-scores, which strike a compromise between recall and accuracy. With a sufficiently high harmonic mean, these numbers indicate that the model successfully balances accuracy and recall for each class. The model does rather well in each class, with accuracy, recall, and F1 scores for each class over 90%. This shows that the model can do the classification job's task of successfully differentiating between the various classes. Taking into consideration class imbalances, the macro average F1-score (95.6%) and weighted average F1-score (95.7%) consider the total performance across all courses. These results show that the model performed well overall on the dataset.

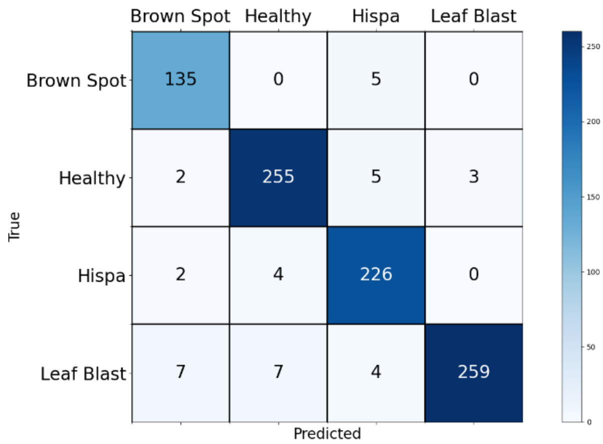

Fig. 7: Confusion Matrix of VGG16 CNN model

Overall, with excellent accuracy, recall, and F1 scores across all classes, the model shows a great ability to categorize examples into the appropriate classes. However, it's crucial to assess the model's effectiveness considering the particular application's and the domain's needs. It may be essential to do further research and testing to determine the model's generalizability and robustness, such as cross-validation or testing on hypothetical data.

Table 2: Result of VGG16 CNN model

	Precision	**Recall**	**F1-Score**	**Support**
BrownSpot	0.964286	0.924658	0.944056	146.00000
Healthy	0.962264	0.958647	0.960452	266.00000
Hispa	0.974138	0.941667	0.957627	240.00000
LeafBlast	0.935018	0.988550	0.961039	262.00000
Accuracy			0.957330	914.000000
Macro Avg	0.958926	0.953380	0.955794	914.00000
Weighted Avg	0.957895	0.957330	0.957259	914.00000

4.2 MobileNet CNN Model

MobileNet is also one of the most used pre-trained CNN models. In this research, the model also promised good results.

4.2.1 Performance of MobileNet CNN Model

In figure 8 and 9, the "Epoch vs Accuracy" graph and the "Epoch vs Validation Loss" graph is depicted. Validation accuracy is increasing according to the increase of epochs and validation loss is decreasing according to the increase of epochs. According to these, it can be said that our MobileNet CNN model is performing well.

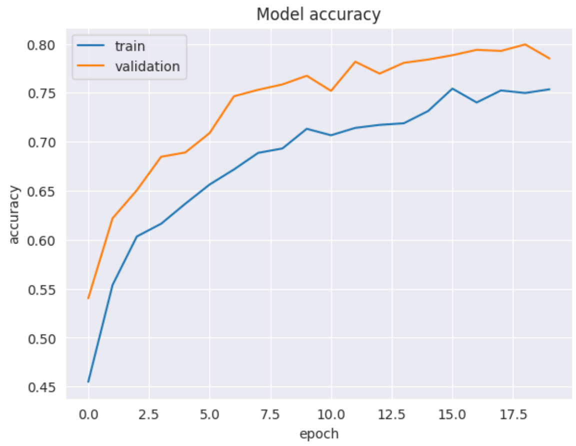

Fig. 8: Epoch vs Accuracy Graph of MobileNet

Fig. 9: Epoch vs Validation Loss Graph of MobileNet

4.2.2 Confusion Matrix of MobileNet CNN model

In Figure 10, the confusion matrix of the MobileNet CNN model is being shown. For the class Brown Spot, 104 are predicted correctly, and 14 are predicted wrong. For the class healthy, 41 are predicted wrong, and for the rest 227 are predicted well. For Hispa, the right prediction number is 201 and the wrong prediction number is 79. For the Life Blast, the right is 222 and the wrong is 26. That means the MobileNet CNN model is performing well but not as well as the VGG16 model.

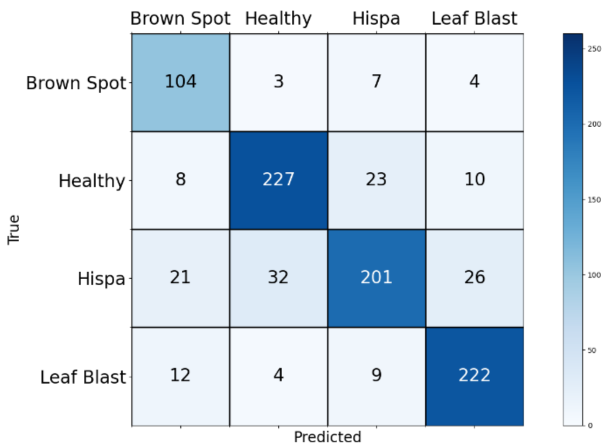

Fig. 10: Confusion Matrix of MobileNet CNN model

4.2.3 Result of MobileNet CNN model

With an overall accuracy of 82.5%, the model successfully predicts the class labels for a significant number of the dataset's cases. Overall, the model shows a respectable capacity for classifying examples into the appropriate groups, with moderate to high accuracy, recall, and F1-scores across several classes (Table 3). The performance, however, might be enhanced even more, particularly for the Hispa class, which has significantly lower accuracy and F1-score than other classes. It is advised to look into and address the causes of the worse performance for that particular class. To determine the generalizability and robustness of the model, further testing and analysis, such as cross-validation or testing on hypothetical data, may be required.

Table 3: Result of MobileNet CNN model

	Precision	Recall	F1-Score	Support
BrownSpot	0.881356	0.712329	0.787879	146.000000
Healthy	0.847015	0.853383	0.850187	266.000000
Hispa	0.717857	0.837500	0.773077	240.000000
LeafBlast	0.895161	0.847328	0.870588	262.000000
Accuracy			0.824945	914.000000
Macro Avg	0.835347	0.812635	0.820433	914.000000
Weighted Avg	0.832387	0.824945	0.825834	914.000000

4.3 ResNet50 CNN Model

It is also one of the most used pre-trained CNN models. It was also used in our research, and it has given a good result.

4.3.1 Performance of ResNet50 CNN Model

The "Epoch vs Accuracy" graph and the "Epoch vs Validation Loss" graph for ResNet50 CNN model is shown in next figure 11 and 12. Validation accuracy represents some ups and downs according to the increase of epochs and validation loss is decreasing according to the increase of epochs. From this, it cannot be concluded that our ResNet50 CNN model is not performing well. Here, the validation accuracy is 46% and the test accuracy is 47.7%. It is not a good result for a machine-learning model.

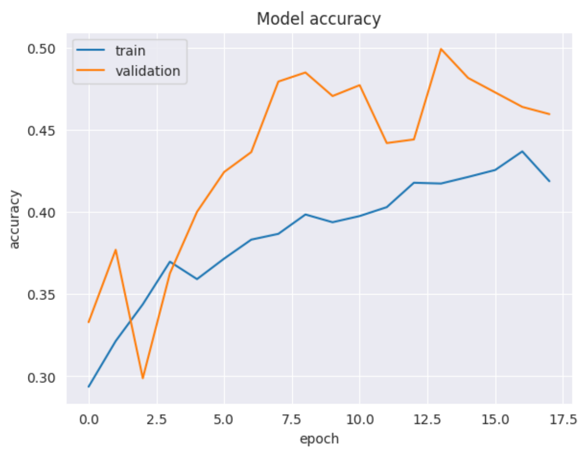

Fig. 11: Epoch vs Accuracy Graph of ResNet50

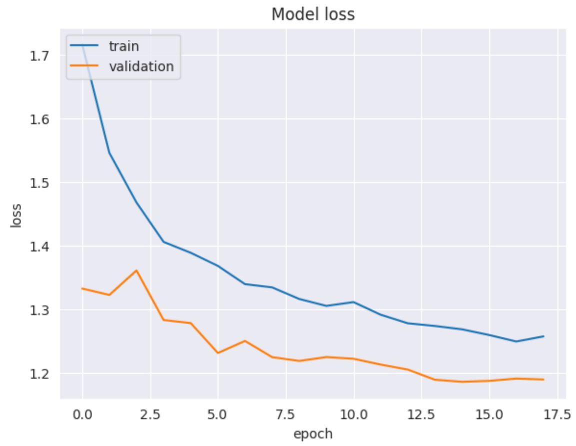

Fig. 12: Epoch vs Validation Loss Graph of ResNet50

4.3.2 Confusion Matrix of ResNet50 CNN model

In Figure 13, the confusion matrix of the ResNet50 CNN model is being shown. For the class Brown Spot, only 15 are predicted correctly, and 9 are predicted wrong. For the class healthy, 288 are predicted wrong, and for the rest 215 are predicted well. For Hispa, the right prediction number is only 60 and the wrong prediction number is 64. For the Leaf Blast, the right is 154 and the wrong is 109. That means the ResNet50 CNN model is not performing well.

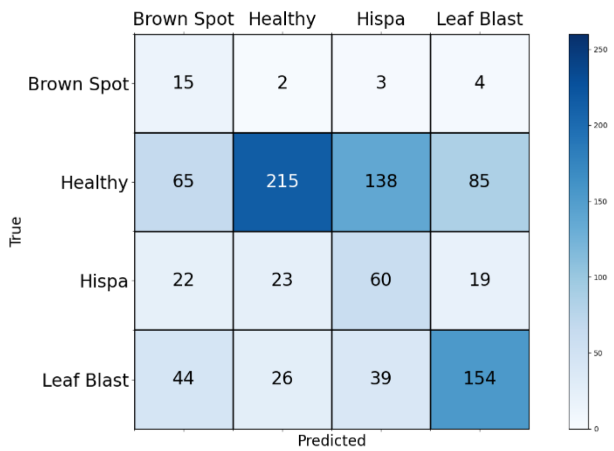

Fig. 13: Confusion Matrix of ResNet50 CNN model

4.3.3 Result of MobileNet CNN model

With an overall accuracy of 47.7%, the model successfully predicts the class labels for a significant number of the dataset's cases. For each class, the precision values vary from 42.74% to 62.5%.

The ResNet50 model's performance varied among the various classes, in conclusion. There is space for improvement when correctly detecting occurrences of the Hispa class, even if it attained greater accuracy for the Brown Spot class and higher recall for the Healthy class. The F1-Score offers a thorough evaluation of the model's overall performance, considering both recall and accuracy. Assessing the model's performance in properly categorizing occurrences of *Oryza sativa* diseases is made easier with an understanding of these assessment indicators.

Table 4: Result of ResNet50 CNN model

	Precision	Recall	F1-Score	Support
BrownSpot	0.625000	0.102740	0.176471	146.000000
Healthy	0.427435	0.808271	0.559168	266.000000
Hispa	0.483871	0.250000	0.329670	240.000000
LeafBlast	0.585551	0.587786	0.586667	262.000000
Accuracy			0.485777	914.000000
Macro avg	0.530464	0.437199	0.412994	914.000000
Weighted Avg	0.519137	0.485777	0.445657	914.000000

4.4 DenseNet121 CNN Model

It is also one of the most used pre-trained CNN models. In this research, this model provided good result.

4.4.1 Performance of DenseNet121 CNN Model

In Figure 14, the "Epoch vs Accuracy" graph and in Figure 15, the "Epoch vs Validation Loss" graph of this model shown. Validation accuracy is increasing according to the increase of epochs and validation loss is decreasing according to the increase of epochs. From this, it can be said that our DenseNet121 CNN model is performing well.

Here, the validation accuracy is 80.82% and the test accuracy is 83.37%. It is a good result for a machine-learning model, but it is not better than the VGG16 model.

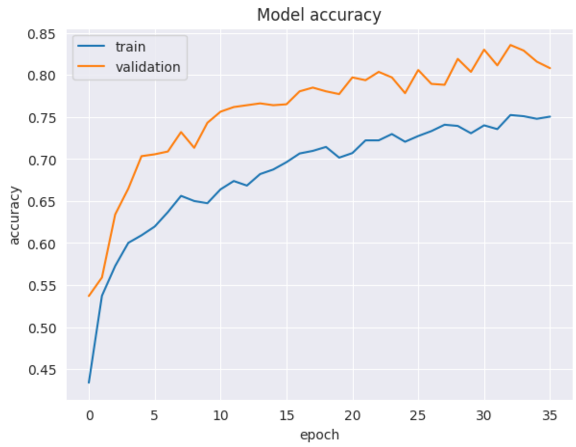

Fig. 14: Epoch vs Accuracy Graph of DenseNet121

Fig. 15: Epoch vs Validation Loss Graph of DenseNet121

4.4.2 Confusion Matrix of DenseNet121 CNN model

In Figure 16, the confusion matrix of the ResNet50 CNN model is shown. For the class Brown Spot, only 15 are predicted correctly, and 9 are predicted wrong. For the class healthy, 288 are predicted wrong, and for the rest 215 are predicted well. For Hispa, the right prediction number is only 60 and the wrong prediction number is 64. For the Leaf Blast, the right is 154 and the wrong is 109. That means the ResNet50 CNN model is not performing well.

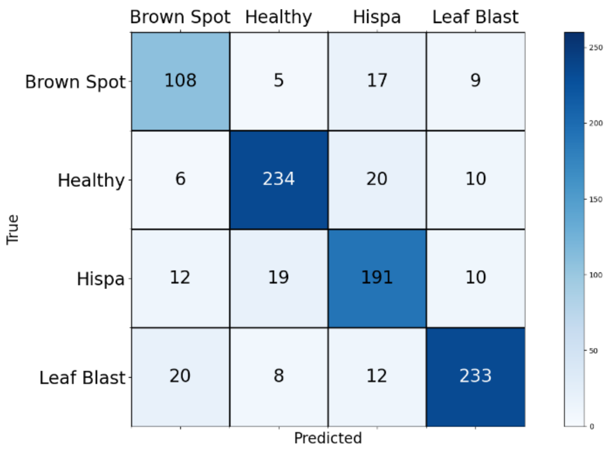

Fig. 16: Confusion Matrix of DenseNet121 CNN model

4.4.3 Result of DenseNet121 CNN model

With an overall accuracy of 83.37%, the model successfully predicts the class labels for a significant number of the dataset's cases. For each class, the precision values vary from 77.69% to 86.67%.

The evaluation metrics for the DenseNet121 model on a multi-class classification job are shown in the Table 5. Brown Spot, Healthy, Hispa, and Leaf Blast were the four classes in the dataset used to train and test the model. The model produced excellent accuracy values across all classes, as shown by the precision column. The Healthy class had the greatest precision score (0.867), meaning that a large percentage of examples were properly classified as belonging to this class. The accuracy ratings for the other classes, such as Brown Spot, Hispa, and Leaf Blast, ranged from 0.776 to 0.853.

The accuracy row displays the model's total accuracy across all classes. In this instance, the model's accuracy was 0.838, meaning that 83.8% of the predictions were accurate. Overall, the DenseNet121 model performs well across all classes, scoring well in terms of accuracy, recall, and F1-Score. Various *Oryza sativa* diseases, such as Brown Spot, Healthy, Hispa, and Leaf Blast, are correctly detected by the model. The usefulness of the DenseNet121 model for disease identification in *Oryza Sativa* plants is shown by the excellent accuracy and balanced F1-Scores.

Table 5: Result of DenseNet121 CNN model

	Precision	**Recall**	**F1-Score**	**Support**
BrownSpot	0.776978	0.739726	0.757895	146.000000
Healthy	0.866667	0.879699	0.873134	266.000000
Hispa	0.823276	0.795833	0.809322	240.000000
LeafBlast	0.853480	0.889313	0.871028	262.000000
Accuracy			0.838074	914.000000
Macro Avg	0.830100	0.826143	0.827845	914.000000
Weighted Avg	0.837166	0.838074	0.837367	914.000000

4.5 User Interface Development (Android Mobile App)

In this research, four pre-trained CNN models have been used. Among all of them, the VGG16 CNN model's accuracy is the highest. An Android mobile app has been developed using the highest performer CNN model VGG16 so that our farmers can directly use the Mobile App for detecting their rice disease.

In the Mobile App, there is only one screen. On this screen, there are a total of three buttons. They are: "Predict Disease", "Take Photo", and "Pick from Gallery".

The functionalities of this mobile app are described below: 1) The farmers will be able to take photos of rice leaves in this App using the "Take Photo" button. 2) The farmers will be able to upload photos in this App from their gallery using the "Pick from Gallery" button. 3) After uploading the photo in the App, they can view it at the top of the screen. 4) Below the photo, farmers will see a button called "Predict Disease". When they click on it, farmers will be able to see the prediction result below.

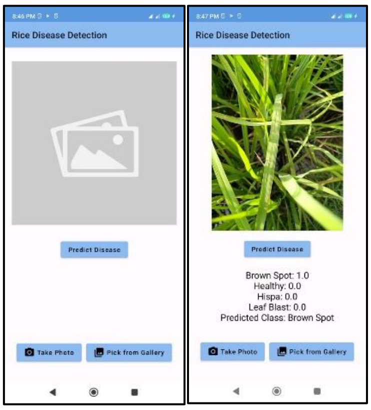

Figure. 17: Demo Screenshots of the Mobile App

5. CONCLUSION AND FUTURE WORKS

5.1 Conclusion

In this research, the efficiency of pre-trained CNN models was investigated that had been adjusted for the detection of *Oryza Sativa* disease. In terms of validation and test accuracy, the performance of four pre-trained models, namely VGG16, MobileNet, ResNet50, and DenseNet121 was distinguished. The pre-trained VGG16 model showed the best level of accuracy, reaching outstanding validation accuracy of 94.5% and test accuracy of 95.3%. This shows that VGG16 performs better than the other models analyzed for *Oryza Sativa* disease detection. The VGG16 model has been successfully integrated into an Android app, making it possible to diagnose diseases quickly and easily on mobile devices. In addition, compared to ResNet50, MobileNet, and DenseNet121, MobileNet had a validation accuracy of 78.5% and a test accuracy of 81.62%, whereas ResNet had a validation accuracy of 46% and a test accuracy of 47.7%. These findings imply that taking use of the information included in pre-trained models considerably improves disease detection skills. Finally, the successful deployment of the VGG16 model in an Android app opens up possibilities for real-time disease detection in the field, providing farmers and agricultural experts with a convenient tool for monitoring and managing *Oryza Sativa* disease outbreaks. The VGG16 model's high accuracy demonstrates its potential for accurate and reliable disease diagnosis, supporting early detection and successful intervention measures.

In conclusion, VGG16 emerges as the best accurate model, demonstrating the effectiveness of tailored pre-trained CNN models for *Oryza Sativa* disease diagnosis. The VGG16 model's effective integration into an Android app offers hope for easy-to-use disease diagnosis in agricultural contexts. Progress in developing disease detection technologies has been made by using deep learning and pre-trained models, which have improved crop health management and increased agricultural output.

5.2 Future Works

Even though this work significantly improved the accuracy of pre-trained CNN models in detecting *Oryza Sativa* rice plant sickness, there are still a number of areas for further research and development needed. In future, the research will expand the dataset and will analysis the various CNN architectures in this field.

6. REFERENCES

[1] P. S. Thakur, T. Sheorey, and A. Ojha, "VGG-ICNN: A Lightweight CNN model for crop disease identification," Multimed Tools Appl, vol. 82, no. 1, pp. 497–520, 2023.

[2] H. Ajra, M. K. Nahar, L. Sarkar, and M. S. Islam, "Disease detection of plant leaf using image processing and cnn with preventive measures," in 2020 Emerging Technology in Computing, Communication and Electronics (ETCCE), 2020, pp. 1–6.

[3] E. L. Mique Jr and T. D. Palaoag, "Rice pest and disease detection using convolutional neural network," in Proceedings of the 1st International Conference on Information Science and Systems, 2018, pp. 147–151.

[4] P. K. Sethy, N. K. Barpanda, A. K. Rath, and S. K. Behera, "Rice false smut detection based on faster R-CNN," Indonesian Journal of Electrical Engineering and Computer Science, vol. 19, no. 3, pp. 1590–1595, 2020.

[5] Z. Saeed, A. Raza, A. H. Qureshi, and M. H. Yousaf, "A multi-crop disease detection and classification approach using CNN," in 2021 International Conference on Robotics and Automation in Industry (ICRAI), 2021, pp. 1–6.

[6] S. Ghosal and K. Sarkar, "Rice leaf diseases classification using CNN with transfer learning," in 2020 IEEE Calcutta Conference (CALCON), 2020, pp. 230–236.

[7] G. Latif, S. E. Abdelhamid, R. E. Mallouhy, J. Alghazo, and Z. A. Kazimi, "Deep Learning Utilization in Agriculture: Detection of Rice Plant Diseases Using an Improved CNN Model," Plants, vol. 11, no. 17, p. 2230, 2022.

[8] K. P. Panigrahi, A. K. Sahoo, and H. Das, "A cnn approach for corn leaves disease detection to support digital agricultural system," in 2020 4th International Conference on Trends in Electronics and Informatics (ICOEI)(48184), 2020, pp. 678–683.

[9] P. A. H. Vardhini, S. Asritha, and Y. S. Devi, "Efficient disease detection of paddy crop using cnn," in 2020 International Conference on Smart Technologies in Computing, Electrical and Electronics (ICSTCEE), 2020, pp. 116–119.

[10] P. Sharma, Y. P. S. Berwal, and W. Ghai, "Performance analysis of deep learning CNN models for disease detection in plants using image segmentation," Information Processing in Agriculture, vol. 7, no. 4, pp. 566–574, 2020.

[11] B. S. Bari et al., "A real-time approach of diagnosing rice leaf disease using deep learning-based faster R-CNN framework," PeerJ Comput Sci, vol. 7, p. e432, 2021.

[12] C. R. Rahman et al., "Identification and recognition of rice diseases and pests using convolutional neural networks," Biosyst Eng, vol. 194, pp. 112–120, 2020.

[13] D. Mohapatra, J. Tripathy, and T. K. Patra, "Rice disease detection and monitoring using CNN and naive Bayes classification," in Soft Computing Techniques and Applications: Proceeding of the International Conference on Computing and Communication (IC3 2020), 2021, pp. 11–29.

[14] G. Zhou, W. Zhang, A. Chen, M. He, and X. Ma, "Rapid detection of rice disease based on FCM-KM and faster R-CNN fusion," IEEE access, vol. 7, pp. 143190–143206, 2019.