Enhancing Paddy Leaf Disease Classification using CNN and MobileNetV2

Rohan Tiwari[1], Neha Vora[2]

[1]Student, [2]Assistant Professor, Department of Information Technology, SVKM's Usha Pravin Gandhi College of Arts, Science and Commerce, Mumbai, India

E-mail: [1]rohantiwari0608@gmail.com, [2]nehavora2501@gmail.com

Abstract

Paddy farming, a cornerstone of global agriculture, faces significant threats from various diseases that affect the crop yield. This research presents a novel approach for detecting paddy leaf diseases using advanced deep learning techniques, specifically transfer learning with the MobileNetV2 architecture. The methodology involves the utilization of a comprehensive dataset consisting of paddy leaf images across multiple disease classes. Data augmentation was extensively employed to address the limitations posed by the dataset size. Both basic and advanced models were trained, with the advanced model achieving a remarkable validation accuracy of 97%. Additionally, Time-Test Augmentation (TTA) was applied to further enhance the model's performance. This research demonstrates the efficacy of deep learning techniques in agricultural disease detection and highlights potential improvements for future applications.

Keywords: Paddy leaf disease, Convolutional Neural Networks, Transfer learning, Data augmentation, MobileNetV2, Image classification.

1. Introduction

Agricultural development is becoming important because the ever-growing population in the world largely depends on agriculture. Almost 90% of people in the world depend on agriculture for their living. About 80% of the global population directly depends on farmers to feed the world. However, the agricultural sector is suffering from heavy losses due to insects, plant diseases, and a variety of economic constraints. These not only affect productivity but also pose a threat to food security in the world at large. Furthermore, there is often the problem

DOI: https://doi.org/10.36548/jscp.2024.3.008
Received: 13.08.2024, received in revised form: 20.09.2024, accepted: 04.10.2024, published: 15.10.2024
© 2024 Inventive Research Organization. This is an open access article under the Creative Commons Attribution-Noncommercial International (CC BY-NC 4.0) License

of a lack of skilled labour and a knowledge deficit in the sector regarding good agricultural practices, including fertilizer use and plant disease control [3].

Rice is one of the most largely consumed staple crops, but it is highly affected by diseases that cause considerable reductions in crop yield. Early symptoms of rice plant diseases often occur on the leaves, so early detection and intervention for the disease are essential for the success of the crop. The conventional system uses on-site physical examination by experts, which can be quite tedious and time-consuming. In this regard, semi-automatic and autonomous disease detection systems have become available, providing fast and accurate options apart from manual observations. These technologies, apart from being very efficient, also help in reducing costs and, therefore, became an important tool for modern agriculture.

Some of the key rice diseases, including rice blast, bacterial blight, and sheath blight, can cause large damage and even lead to huge economic losses. For example, Magnaporthe oryzae produces rice blast with lesions on leaves, stems, and panicles, seriously affecting crop yields. Other diseases, such as bacterial blight (Xanthomonas oryzae), cause the rice leaves to turn yellowish and wilt, which also adds up to a huge loss in production. Apart from these, diseases like rice tungro, brown spot, and false smut are also the prime enemies of rice cultivation and hence there is a dire necessity for a highly effective system for disease detection.

This research investigates the use of deep learning techniques in general, and particularly the use of Convolutional Neural Networks for automating the process of detecting rice plant diseases. The research suggests an efficient and reliable way of diagnosing such crop diseases using advanced image classification models, such as MobileNetV2, incorporated with data augmentation techniques. Thus, contributing to improvement in productivity and sustainability in agriculture through automation in disease detection and related fields.

2. Related Work

Deep learning has been the breakthrough in computer vision, allowing machines to see and recognize objects with incredibly high accuracy across image recognition classification. The most important development took place with the release of AlexNet by Krizhevsky during the 2012 ImageNet competition, creating a breakthrough in image classification. Since then, multiple architectures have been introduced (e.g., VGG, ResNet, and Inception), with architectural modifications on top of convolutional neural networks (CNNs) to increase performance and recall extraction efficiency for CNNs. These models have been invaluable for solving complex image recognition challenges spanning multiple industries, not least in agriculture [1], [5].

Transfer learning is a very powerful technique, especially in cases of limited or small labeled datasets. MobileNetV2 [2] is developed specifically for mobile and embedded systems, providing better speed and accuracy for classification. This architecture is computationally efficient due to depthwise separable convolutions. Transfer learning enables researchers to take existing models like MobileNetV2 and adapt them for custom tasks by retraining the model on smaller, domain-specific datasets. This has significantly reduced training time and improved model performance for specific tasks like the classification of crop diseases [4], [6]. There has been extensive research on employing CNNs for plant disease detection. Ferentinos [15] also showed that CNNs in plant disease diagnosis performed well even with smaller datasets using transfer learning. Various architectures such as ResNet and Inception have been successfully adapted to CNNs, achieving high classification accuracy in agriculture [7], [8].

However, although great success has been achieved in this area, research on deep learning for paddy leaf disease detection is still relatively limited. Several studies have utilized machine learning to identify paddy diseases, but these are mainly traditional optimization-based solutions such as image processing or traditional classifiers, which lack the impact of deep learning models [1], [9]. Furthermore, the datasets used in these studies are often limited in dimensionality and variability [10], [11].

This research aims to address these limitations by employing a deep learning approach, utilizing MobileNetV2 in conjunction with data augmentation techniques, to improve the classification accuracy of paddy leaf diseases. By utilizing these advanced methodologies, this research seeks to contribute to the growing field of automated disease detection in agriculture [12].

3. Methodology

3.1 Dataset and Preprocessing

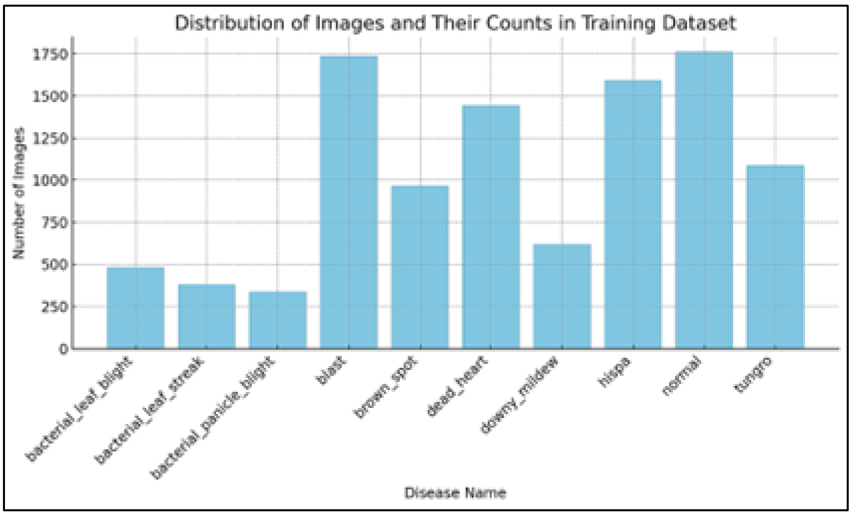

Figure 1. Distribution Images and their Counts in Training Dataset

The secondary dataset used for this study consists of 10 class paddy leaf diseases as shown in Table 1. The dataset contained 10,407 images for training and 3,469 test images spanned over Bacterial Leaf Blight (BLB), Blast, Brown Spot (BS), Dead Heart (DH), tungro as shown in Figure 1. The images were resized to 224x224 pixels which is the average input size for most of the pre-trained models such as MobileNetV2.

The images underwent several augmentation techniques (Figure 2) to enhance the dataset. Random rotations were applied to the images, with a stochastic rotation of up to 10 degrees. Zoom transformations were also introduced, using a fixed Local Linear Transformation (LLT) with a zoom range of 0.1, randomly applied while reading data samples from the dataset. Geometric distortions were incorporated through shear transformations with a factor of 0.25. Additionally, the images were flipped both horizontally and vertically to ensure that their orientation had no significant impact, regardless of how they were captured. Finally, translation transformations were used, applying small displacements in width and height up to 10%, adding variability to the training dataset.

These transformations ensured that the model would be trained on a diverse set of images and could give accurate classification in real-life scenarios.

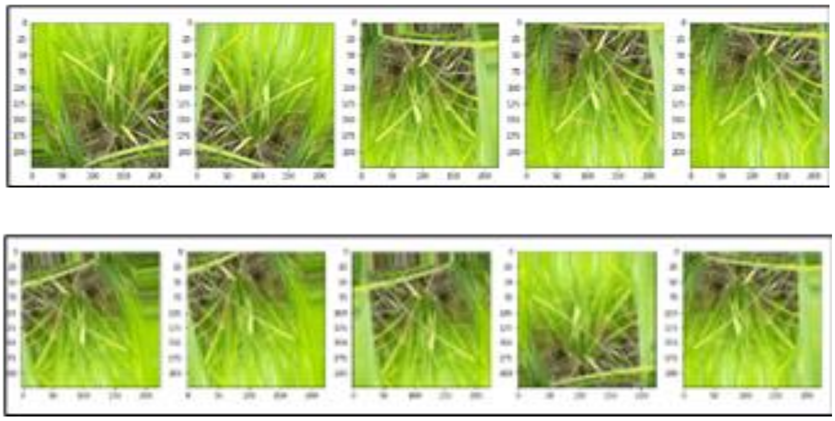

Figure 2. Data Augmentation on Images

Table 1. Training Images Summary

Folder Name	Number of Files	Description
bacterial_leaf_blight	479	Images of paddy leaves affected by bacterial leaf blight disease.
bacterial_leaf_streak	380	Images showing bacterial leaf streak on paddy leaves.
bacterial_panicle_blight	337	Photos of paddy plants with bacterial panicle blight disease.
blast	1738	Paddy leaves showing blast disease symptoms.
brown_spot	965	Images of paddy leaves affected by brown spot disease.
dead_heart	1442	Photos displaying dead heart disease in paddy plants.
downy_mildew	620	Paddy plants suffering from downy mildew infection.
hispa	1594	Images of paddy plants affected by hispa disease.
normal	1764	Healthy paddy leaves without any disease symptoms.
tungro	1088	Photos of paddy plants infected by tungro virus.

3.2 Model Architecture

3.2.1 Proposed Model

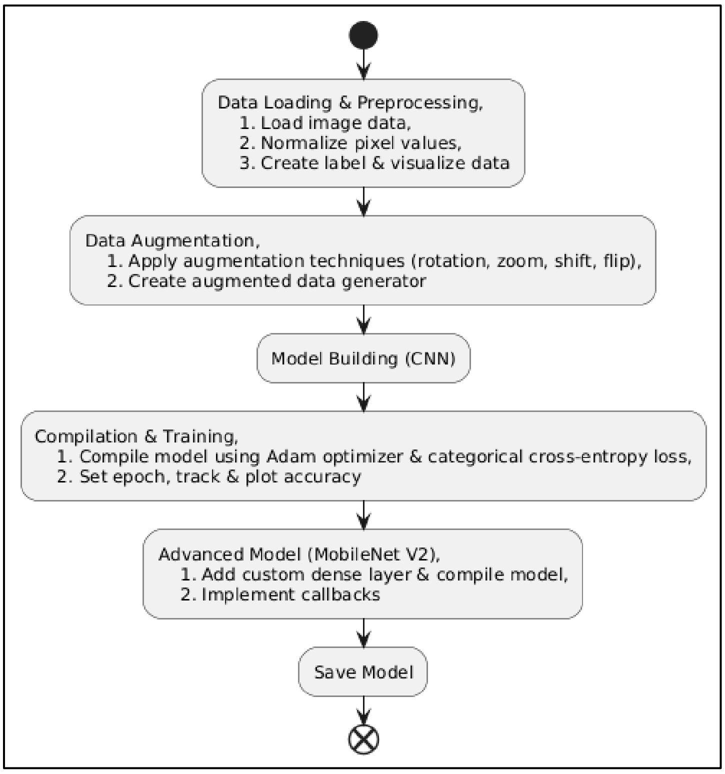

Figure 3. Workflow of Paddy Leaf Disease Classification

The process begins with data loading and preprocessing (Figure 3), where image data is loaded, pixel values are normalized, and labels are created for visualization purposes. Following this, the data augmentation step involves applying various augmentation techniques such as rotation, zoom, shift, and flip to the images, and generating an augmented data set to enhance the model's training.

Next, in the model-building phase, a CNN architecture is created. The compilation and training steps are as follows, where the model is compiled using the Adam optimizer and categorical cross-entropy loss. The number of epochs is set, and the model's accuracy is tracked and plotted during training. Afterward, an Advanced Model (MobileNet V2) is employed, where a custom dense layer is added, and the model is recompiled, with callbacks implemented to enhance performance. Finally, the model is saved for future use.

3.2.2 Basic CNN Model

The model begins with an input layer that accepts images in the shape of 224x224x3, where 3 represents the RGB channels. The first convolutional layer applies 32 filters of size 3x3, designed to detect low-level features such as edges and corners. This layer uses the ReLU activation function, which introduces non-linearity, allowing the model to learn more complex representations. Following this, a max-pooling layer with a pool size of 2x2 is used to down-sample the feature maps, retaining only the most dominant features.

Next, a second convolutional layer with 64 filters of size 3x3 builds upon the features detected by the first layer, focusing on more complex patterns, including texture and shape differences. This is followed by another 2x2 max-pooling layer, which further reduces the spatial dimensions of the features, improving computational efficiency. The third convolutional layer, consisting of 128 filters with a 3x3 kernel size, detects even more complex features, such as those specific to plant diseases. This is again followed by a 2x2 max-pooling layer to prepare the features for the fully connected layers.

The output of the convolutional layers is flattened and passed to a fully connected layer with 1,024 neurons, using the Swish activation function, which offers a smoother transition than ReLU and enhances performance in some tasks. The final layer is a softmax layer with 10 neurons, one for each disease category. Softmax is applied to generate a probability distribution, and the model makes its final prediction based on the highest probability. The basic architecture of the CNN is like (Shown in Figure4)

Here we have the trained basic CNN model for 10 epochs which has a validation accuracy of around ~79% and hence, that means — to achieve higher accuracies begin advanced approaches. The efficacy to recognize visual patterns in paddy leaf diseases reflects the value of CNNs which can indeed be inferred easily by a simple basic model.

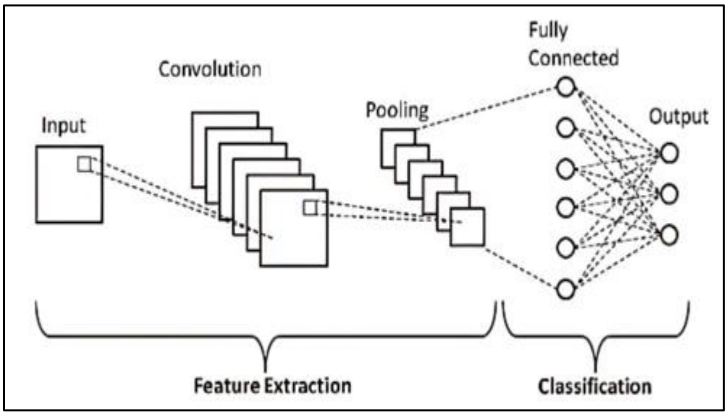

Figure 4. The Basic Architecture of the CNN

3.2.3 Transfer Learning with MobileNetV2

The limitations of the simple CNN model led us to investigate transfer learning with MobileNetV2 (Figure 5), an efficient and accurate architecture. It is because models pre-trained on large datasets (like ImageNet) can be fine-tuned to other specific tasks, the technique known as transfer learning, which considerably reduces training effort after all. MobileNetV2 Neuronal Network Architecture – Mobile Nets are lightweight deep neuronal network architecture particularly well suited for mobile and resource-constrained environments. Here is how it looks, we use depth-wise separable convolutions to reduce the number of parameters without loss in performance. For the transfer learning model, the base network was chosen as MobileNetV2 which has been pre-trained. The model represents the first initial layers that help to recognize general image features (edges, textures) that were cut off. The layers are very transferable across domains as they capture basic visual information. Some of the latter, more ImageNet-specific layers were removed and replaced with new fully connected layers customized to predict classes in the rice disease dataset.

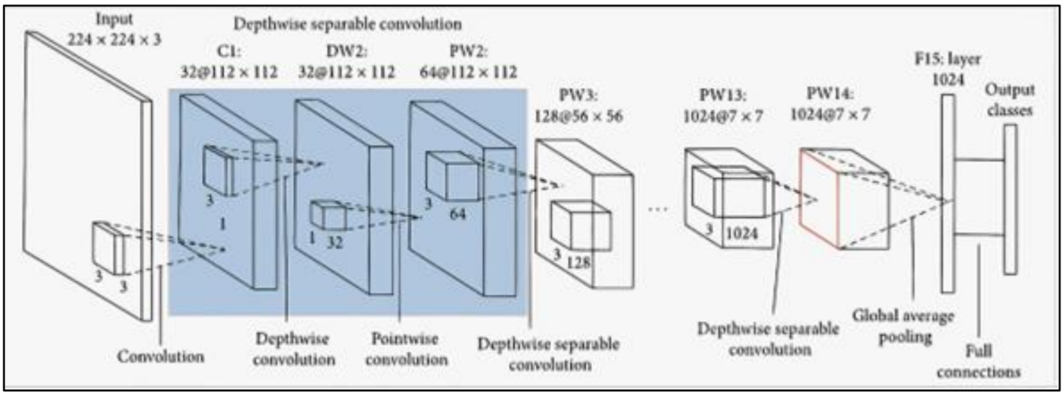

Figure 5. MobileNetV2 Architecture

3.2.4 New Architecture for Transfer Learning

The model starts with the MobileNetV2 base, which is integrated with pre-trained weights from ImageNet. This base is not retrained, allowing other layers to be fine-tuned, and utilizes depth-wise separable convolution layers to reduce model size and computational cost. Instead of flattening the entire feature map, a global average pooling layer is used to condense each feature map into scalar values, preserving spatial information while converting the output into a 1D tensor. This step allows for the addition of dropout and batch normalization, which help regularize the input.

Following this, a fully connected dense layer with 1,024 units and the Swish activation function learns complex feature combinations detected by MobileNetV2. The Swish activation function, known for its smooth non-linearity, enables faster model convergence compared to ReLU. A second dense layer with 128 units further refines the feature representations before classification. Finally, the output layer classifies the images into one of the 10 disease categories using a SoftMax layer with 10 units.

The below-shown architecture (Figure6) was designed to handle the dataset-specific instead of its generalized abilities to handle any dataset. This was done by freezing the initial layers to avoid the problem of overfitting. This allowed to develop new layers whose sole purpose was to focus on disease-specific visual pattern in the rice leaf image.

Layer (type)	Output Shape	Param #
resizing_1 (Resizing)	(None, 224, 224, 3)	0
mobilenetv2_1.00_224 (Functional)	(None, 1280)	2257984
dense_3 (Dense)	(None, 1024)	1311744
dense_4 (Dense)	(None, 128)	131200
dense_5 (Dense)	(None, 10)	1290

Figure 6. Modified Architecture for MobileNetV2

3.3 Experimental Setup

Table 2 below shows the hardware and the software used in the work

Table 2. Hardware and Software Used

Language	Python3
Processor	T4 GPU
RAM	16GB
Storage	128GB
Operating System	Ubuntu
Environment	Google Collab
Framework	TensorFlow, Keras
Pre-Trained Model	MobileNetV2

3.4 Training Procedure

The model built was compiled with an Adam optimizer at 0.001 learning rate and categorical cross-entropy loss. Early stopping, a powerful training strategy is utilized to halt the training process whenever the validation loss does not improve for 8 epochs, thereby preventing overfitting. To help stabilize the training, batch normalization was utilized between all fully Connected layers (except for predictions) and uniform dropout with rate 0.5 was used on top of fully connected layers to mitigate overfitting problems. The model was checkpointed with respect to the validation accuracy. The model was trained up to 100 epochs using a batch size of 32 and with the help of 16 workers raising data preprocessing speed.

Table 3 below highlights key epochs where significant performance changes were observed during the training process. It includes the accuracy, loss, and adjustments to the learning rate for each epoch, providing insights into model optimization.

Table 3. Highlight of Key Epoch

Epoch	Training Accuracy	Validation Accuracy	Training Loss	Validation Loss	Learning Rate
1	57%	13%	1.79	1.7	0.001
10	90%	78%	0.74	0.79	0.001

20	94%	87%	0.33	0.42	0.001
40	97%	97%	0.19	0.22	0.001
46	98%	97%	0.15	0.14	0.0004
52	99%	97%	0.12	0.16	0.0001

3.5 Test Time Augmentation

Time-Test Augmentation (TTA) was performed to improve the accuracy of the model during evaluation. TTA consists of augmenting each test image several times (flips, rotations...) and averaging the predictions of the model over these augmented images. Such regular Gaussian noise resulted in better predictions since the model experienced not only slightly varied input images by also more dense domains. During different training processes, it could be called with a new set of randomly enqueued augmented data.

After applying TTA during the evaluation phase, the model's accuracy increased from 96.6% to 97.0%, demonstrating the method's ability to provide a marginal yet remarkable improvement in performance. This increase, although small, is significant in tasks where the cost of misclassification is high.

4. Results

The performance of both Basic CNN and the transfer learning model was tested using validation set. Various metrics like accuracy, loss, and validation accuracy were recorded to assess the model’s efficacy in detection of paddy leaf diseases.

4.1 Basic CNN Model Performance

CNN basic model which was the first baseline was trained for 10 epochs. The model had a training accuracy of 79%, despite its simplicity and the fact that this number was higher than expected, the validation score settled around 70%. The accuracy of this model successfully distinguished certain disease families but was not as successful, especially in surface lesions with similar symptoms for instance the two bacterial blights; leaf and panicle where misclassifications were more common.

The basic model had fewer parameters and was less able to learn complex patterns in the data. It will also overfit faster as the dataset is small. This can be observed from the increasing gap between training and validation accuracy. Even though the CNN model can detect simple diseases it was not robust enough to classify diseases that needed fine-grained analysis. The Table 4 Summarize the basic CNN Model Performance.

Table 4. Performance of Basic CNN Model

Epoch	Training Accuracy	Validation Accuracy	Training Loss	Validation Loss	Note
1	57%	13%	1.79	1.7	Initial epoch; poor Performance
5	75%	58%	1.1	1.2	Significant Improvement in validation.
10	79%	70%	0.9	0.95	Model starts to plateau
15	82%	68%	0.85	0.85	Slight overfitting Observed.

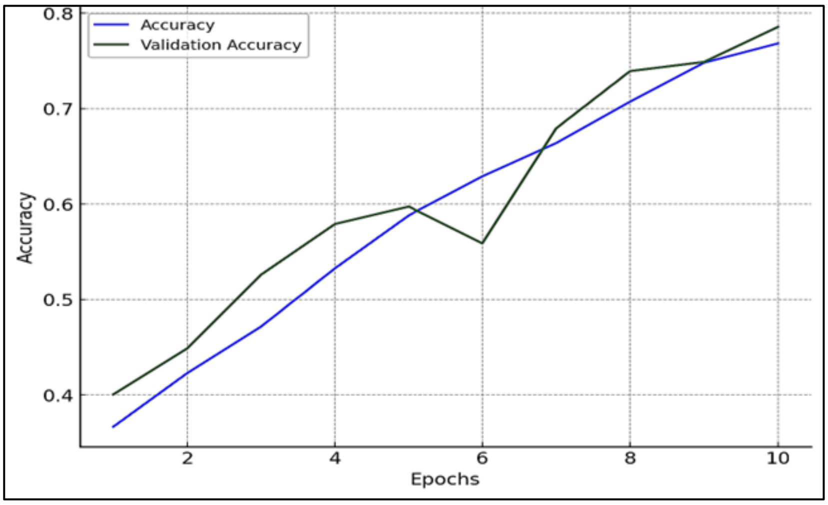

Figure 7. Accuracy vs Validation Accuracy (CNN)

In Figure 7 Validation accuracy crosses Accuracy multiple times which indicates the model is unstable. The model's weight changes too drastically which is leading to inconsistent performance between training and validation accuracy.

4.2 MobileNetV2 Transfer Learning Model (Advanced Model)

Using transfer learning with MobileNetV2 led to a substantial improvement in classification performance. The advanced model outperformed the accuracy of the basic CNN within 10 epochs and achieved an accuracy of 90% and a validation accuracy of 78%. The advanced model continues to show improvement which each epoch and achieved an accuracy of 97% by the end of the 40th epoch. The improvements where likely due to a pre-trained model of MobileNetV2 which had high feature extraction capabilities. The Table 5 shows Key Epoch Metrics.

Table 5. Key Epoch Metrics for Advanced Model

Epochs	Training Accuracy	Validation Accuracy	Training Loss	Validation Loss
1	57%	13%	1.79	1.7
10	90%	78%	0.74	0.79
20	94%	87%	0.33	0.42
40	97%	97%	0.19	0.22
46	98%	97%	0.15	0.14

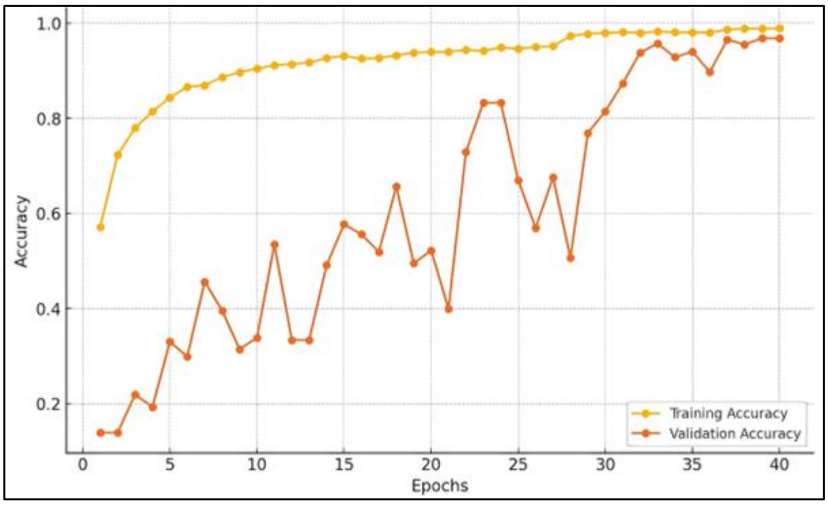

Figure 8. Training Accuracy vs Validation Accuracy (Advanced Model)

Figure 8 shows the steady merger of training accuracy with validation accuracy. This means the model is learning well and that the training process is stable. It is an indicator that the learning rate, model architecture, and other hyperparameters are appropriately set.

4.3 Classification Metrics

Table 6 shows the precision, recall, and F1 scores of various diseases in the dataset.

Table 6. Classification Performance Metrics

Diseases Class	Precision	Recall	F1-Score
Bacterial Leaf Blight	97%	96%	96.50%
Bacterial Leaf Streak	98%	99%	98.50%
Bacterial Panicle Blight	96%	94%	95%
Blast	98%	98%	98%
Brown Spot	97%	96%	96.50%
Dead Heart	96%	95%	95.50%
Downy Mildew	98%	97%	97.50%
Hispa	97%	98%	97.50%
Normal	99%	99%	99%
Tungro	98%	97%	97.50%
Overall	97%	97%	97%

4.4 Time Test Augmentation Results

Time-test augmentation was applied to enhance the robustness of the model prediction. TTA increased the accuracy marginally by averaging predictions on multiple augmented versions of each test image. TTA improved overfitting, increasing validation accuracy to 97.2%, adding further evidence that TTA can improve the ability of models — specifically on much nuance cases of visually ambiguous or difficult-to-distinguish diseases present in pixels. The disparity between the basic CNN and MobileNetV2 models clearly demonstrated the benefits of transfer learning while helping us understand our model architectures. Basic CNN validation accuracy is 70% and for MobileNetV2 model it is around 97%. This improvement exemplifies the necessity of deploying state-of-the-art architectures and pre-trained models, especially when dealing with real-time images.

5. Conclusion

The study proposed a fast and reliable method for detecting paddy leaf diseases using deep learning with transfer learning on MobileNetV2. This approach significantly improved accuracy, achieving ~97.2% in a 5-class validation, compared to just 70% using a basic CNN. Transfer learning, utilizing a pre-trained ImageNet model, enabled precise plant disease identification. Data augmentation helped mitigate overfitting, while Test Time Augmentation (TTA) further improved prediction reliability. This work demonstrates how deep learning can transform agricultural disease detection by offering a low-cost, high-accuracy, and fast solution, reducing the need for manual crop inspections. Future work could involve deploying the model on edge devices for real-time detection and expanding it to cover more diseases under varied environmental conditions for broader applicability.

References

[1] Saha, S., Ray, H., Ghosh, A., Chakraborty, A., Ghosh, D., & Ghosh, T. (2021, May). “Identification of Paddy Leaf Disease (Blast and Brown Spot) Detection Algorithm”. In 2021 2nd International Conference on Secure Cyber Computing and Communications (ICSCCC)Jalandhar · India, IEEE. 23-28

[2] Rao, R. "Jaya: A simple and new optimization algorithm for solving constrained and unconstrained optimization problems." International Journal of Industrial Engineering Computations 7, no. 1 (2016): 19-34.

[3] Kerkvliet, B. J. T. (2018). Vietnam's rural transformation. Routledge.https://api.pageplace.de/preview/DT0400.9780429971815_A36243774/preview-9780429971815_A36243774.pdf

[4] Pantazi, X. E., Moshou, D., & Tamouridou, A. A. (2019). “Automated leaf disease detection in different crop species through image features analysis and One Class Classifiers. Computers and electronics in agriculture”, 156, 96-104.

[5] Swathika, R., Srinidhi, S., Radha, N., & Sowmya, K. (2021, February). “Disease Identification in paddy leaves using CNN based Deep Learning”. In 2021 third

international conference on intelligent communication technologies and virtual mobile networks (ICICV)Tirunelveli, India, IEEE. 1004-1008

[6] Atila, Ü., Uçar, M., Akyol, K., & Uçar, E. (2021). “Plant leaf disease classification using EfficientNet deep learning model”. Ecological Informatics, 61, 101182.

[7] Verma, Ruchi, and Varun Singh. "Leaf disease identification using densenet." In International conference on artificial intelligence and speech technology, pp. 500-511. Cham: Springer International Publishing, Delhi, India, 2021.

[8] Wang, Yibin, Haifeng Wang, and Zhaohua Peng. "Rice diseases detection and classification using attention based neural network and bayesian optimization." Expert Systems with Applications 178 (2021): 114770.

[9] Zakzouk, Salma, Mohamed Ehab, Silvana Atef, Retaj Yousri, Rania M. Tawfik, and M. Saeed Darweesh. "Rice leaf diseases detector based on AlexNet." In 2021 9th international Japan-Africa conference on electronics, communications, and computations (JAC-ECC), Alexandria, Egypt. IEEE, 2021. 170-174.

[10] Upadhyay, Santosh Kumar. "Rice leaves disease detection and classification using transfer learning technique." In 2022 2nd International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE), Greater Noida, India. IEEE, 2022. 2151-2156.

[11] Patil, Rutuja R., and Sumit Kumar. "Rice-fusion: A multimodality data fusion framework for rice disease diagnosis." IEEE access 10 (2022): 5207-5222.

[12] Prajapati, Harshadkumar B., Jitesh P. Shah, and Vipul K. Dabhi. "Detection and classification of rice plant diseases." Intelligent Decision Technologies 11, no. 3 (2017): 357-373.

[13] Petchiammal, Briskline Kiruba, Murugan, and Pandarasamy Arjunan. "Paddy doctor: A visual image dataset for automated paddy disease classification and benchmarking." In Proceedings of the 6th Joint International Conference on Data Science & Management of Data (10th ACM IKDD CODS and 28th COMAD), Mumbai, India. 203-207. 2023.

[14] Bharathi, R. J. (2020). Paddy plant disease identification and classification of image using AlexNet model. International Journal of Analytical and Experimental Modal Analysis12(0886), 1094-1098.

[15] Ferentinos, Konstantinos P. "Deep learning models for plant disease detection and diagnosis." Computers and electronics in agriculture 145 (2018): 311-318.

Author's biography

Rohan Tiwari is a recent graduate with a Bachelor of Science in Information Technology, achieving an impressive CGPA of 9.67. With a strong academic foundation, Rohan's core interests lie in the fields of Artificial Intelligence (AI) and app development. He is passionate about leveraging AI technologies to create innovative solutions, and his expertise in app development further complements his technical skills.

Neha Vora is currently pursuing her Ph.D. in Computer Science and holds a master's in computer applications (MCA). She is qualified in NET, SET, and GATE, and brings over 9 years of teaching experience, along with 1 year of industry experience. Her primary research areas include computer vision, image processing, machine learning, object detection, and artificial intelligence.