Discover Artificial Intelligence

Research

Rice Plant Disease Detection using Convolutional Neural Networks

A. Bala Ayyappan[1] · T. Gobinath[2] · M. Kumar[3] · A. Sivaramakrishnan[4]

Received: 12 November 2024 / Accepted: 25 April 2025
Published online: 09 May 2025

Abstract
Accurately identifying paddy diseases is essential for achieving optimal and quality yield. However, the traditional method of identifying diseases depends on human expertise, and it is prone to errors. In this paper, we use Convolutional Neural Networks (CNNs) and deep learning approaches to identify various rice plant diseases like blast, brown spot and bacterial blight. The CNN model is trained on images of different plant diseases, and various models are evaluated to determine the most effective one for disease identification. The findings of this research will contribute to automated paddy disease diagnosis, aiding farmers in timely and effective disease management. The various models attained different accuracies in paddy disease classification: DenseNet121 achieved an accuracy of 97.50%, the Xception algorithm achieved 96.32%, EfficientNet B4 achieved 96.25%, and MobileNet V3 Large also achieved 96.25%.

Keywords Convolutional neural networks · Deep learning · Plant disease detection

1 Introduction

Paddy is one of the most crucial crops in global food production, and it is prone to several diseases that can significantly reduce optimal and quality yield. The timely detection and identification of plant diseases is crucial for effective disease control. Traditionally, agricultural experts rely on visual inspections to identify crop diseases. However, this approach is often costly, time consuming, and susceptible to human error, which can result in delays in diagnosis and treatment. Computer vision and deep learning techniques, particularly Convolutional Neural Networks (CNNs), have demonstrated strong performance in detecting and classifying plant diseases by recognizing patterns and features in images. In recent years, researchers have applied CNN-based methods to classify paddy diseases such as blast, brown spot, and bacterial blight by training models on large image datasets.This work aims to develop a CNN model trained on diverse paddy disease datasets [1] and compare the performance of different image classification models to automate the identification of diseases in paddy crops. By integrating CNN models with high-resolution imagery of paddy crops, we seek to create a robust system for early disease identification, which will ultimately contribute to improving crop health and yield.

A. Bala Ayyappan, T. Gobinath and A. Sivaramakrishnan have contributed equally to this work.

✉ M. Kumar, mkumar.ece@gmail.com; A. Bala Ayyappan, balaayyappan@hotmail.com; T. Gobinath, gobinath19@gmail.com; A. Sivaramakrishnan, sivainccet@gmail.com | [1]Artificial Intelligence and Data Science, Chettinad College of Engineering and Technology, NH67, Karur-Trichy Highway, Puliyur CF PO, Karur 639114, Tamilnadu, India. [2]Computer Science and Engineering, Chettinad College of Engineering and Technology, NH67, Karur-Trichy Highway, Puliyur CF PO, Karur 639114, Tamilnadu, India. [3]Electronics and Communication Engineering, Chettinad College of Engineering and Technology, NH67, Karur-Trichy Highway, Puliyur CF PO, Karur 639114, Tamilnadu, India. [4]Artificial Intelligence and Data Science, Chettinad College of Engineering and Technology, NH67, Karur-Trichy Highway, Puliyur CF PO, Karur 639114, Tamilnadu, India.

2 Literature review

Bacterial Leaf Blight is a significant disease affecting paddy crops, affected by the bacterium Xanthomonas oryzae pv. oryzae. It primarily affects rice plants, leading to severe economic losses in rice production. In advanced stages, the disease can cause complete leaf blight, severely reducing photosynthesis and crop yield [2]. Bacterial Leaf Streak is another disease affecting paddy rice, triggered by the bacterium Xanthomonas oryzae pv. oryzicola. This disease is less severe compared to bacterial leaf blight but can still significantly impact rice crops [3].Bacterial Panicle Blight is a serious disease affecting the rice plants, resulting from the bacterium Burkholderia glumae. This disease primarily targets the panicles, which are the flower-bearing structures of the rice plant and can lead to significant losses in rice yield and quality [4]. Blast is a major and destructive disease affecting rice crops, due to the fungus Magnaporthe oryzae (formerly known as Pyricularia oryzae). It is one of the most economically significant diseases affecting the rice cultivation worldwide [5]. Brown Spot is a disease affecting rice plants, induced by the fungus Bipolaris oryzae (formerly known as Cochliobolus miyabeanus). It is a significant disease in rice cultivation, particularly in regions with high humidity [6]. Downy Mildew is a fungal disease affecting various crops, including rice, caused by the pathogen Peronosclerospora spp. It is characterized by a range of symptoms that can significantly impact crop health and yield [7]. Hispa refers to a group of insect pests that affect rice crops, particularly in Asia. The most notable of these pests is the rice hispa beetle, scientifically named Dicladispa armigera. Heavy infestations can lead to leaf curling and distortion, impacting plant growth and development [8].This implies that the plants are healthy, growing as expected, and showing no symptoms of disease or pest infestation. Tungro is a severe viral disease affecting rice crops due to complex of two viruses: The Rice Tungro Spherical Virus (RTSV) and the Rice Tungro Filamentous Virus (RTFV). It causes significant problem in rice cultivation, especially in Asia [9]. Convolutional Neural Networks (CNNs) and Vision Transformers (ViTs) are at the forefront, excelling in classification, detection, and segmentation tasks. Public datasets like PlantVillage and custom datasets have facilitated advancements, with models like DenseNet, EfficientNet, and hybrid CNN-ViT architectures achieving accuracies exceeding 99% in some cases. These technologies enhance early disease detection, sustainability, and productivity in agriculture [10]. The focus on grape leaves highlights the utility of CNNs and ViTs in diagnosing diseases like Black Rot, Leaf Blight, and Esca, achieving perfect accuracy in some models like Swinv2-Base. Using datasets such as PlantVillage and Grapevine, deep learning models have demonstrated exceptional performance in disease detection and leaf classification, with pre-trained models fine-tuned to maximize accuracy. The challenges like dataset diversity, real-world generalization, and scalability pave the way for future research integrating multimodal data and real-time monitoring for broader applicability [11].The effectiveness of ViTs in classifying diseases such as apple leaf scab, black rot, and cedar apple rust has been demonstrated with high accuracy [12]. The study [13] demonstrates the potential of CNN-based deep learning models for paddy leaf disease detection, specifically using Inception-ResNet-V2. By leveraging transfer learning, this approach achieves high accuracy while reducing training complexity.The CNNIR-OWELM (Convolutional Neural Network-based inception with ResNet v2 and Optimal Weighted Extreme Learning Machine) model uses IoT devices to capture rice plant images, preprocesses them with Contrast Limited Adaptive Histogram Equalization (CLAHE) for enhanced contrast, segments affected regions via histogram-based methods, extracts features using Inception with ResNet v2, and classifies diseases using the Flower Pollination Algorithm [14]. Li.R et.al used models like VGG16, ResNet, DenseNet, and MobileNet for rice disease detection. For instance, VGG19 achieved training accuracies of around 77% in some cases, while MobileNet reached approximately 76.9% under similar conditions. A noteworthy implementation of ResNet variants for a smartphone application achieved 91% accuracy [15].Haridasan A. et al. combined computer vision and machine learning techniques to automate disease identification using Sobel edge detection and K-means clustering [16].

3 Methodology

3.1 Class weight

The Class Weight technique is used to handle class imbalance in machine learning models. Class imbalance occurs when the classes in a dataset are not represented equally, often leading to poor model performance in the minority class. The Class Weight technique addresses this by assigning different weights to classes, ensuring that the model gives more importance to the minority class during training [17].The dataset consists of 10,407 images across 10 classes. Table 1 gives the weight for each classes.

Table 1 Class weights for Paddy Dataset

Class	Weight
Bacterial_leaf_blight	217.27
Bacterial_leaf_streak	273.87
Bacterial_panicle_streak	308.81
Blast	59.88
Brown_spot	107.84
Dead_heart	72.17
Downy_mildew	167.85
Hispa	65.29
Normal	59.00
Tungro	95.65

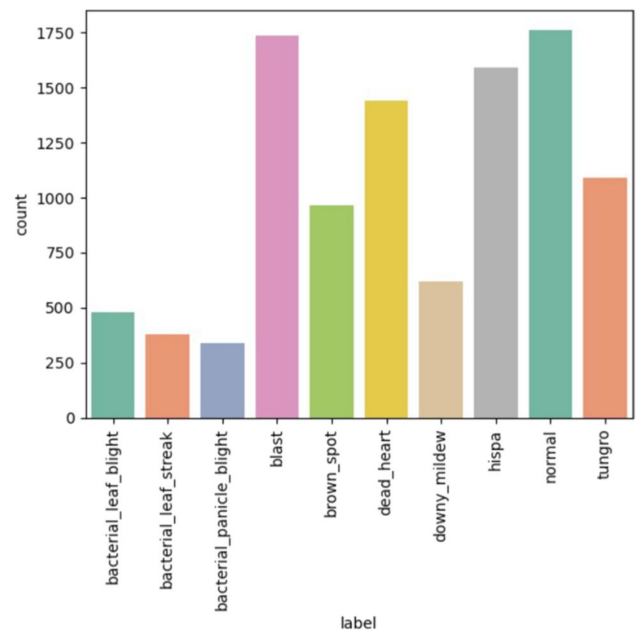


Fig. 1 Imbalanced dataset

In Fig. 1, the imbalanced dataset shows that the classes "normal" and "blast" have the highest frequencies, with only a minor difference between them, mainly due to their larger sample sizes. In contrast, the "bacterial_panicle_blight" class has the lowest frequency.

Figure 2, however, illustrates the Class Weight graph, which depicts the opposite situation. Here, "bacterial_panicle_blight" has a higher weight due to its low frequency, while "blast" has a lower weight because of its higher frequency. This contrast emphasizes the differing effects of class weighting between the two figures.

3.2 Convolutional neural networks (CNNs)

CNNs are dominant tool in machine learning, particularly in jobs related to image processing. CNNs are specific class of neural networks intended to efficiently process grid-like data, such as images. A CNN is a deep learning algorithm that excels in image recognition and processing tasks. It is composed of multiple layers, including convolutional layers, pooling layers, and fully connected layers. CNNs are designed based on the principles of visual processing in the human brain, which makes them particularly adept at detecting hierarchical patterns and spatial relationships in images [18]. Various types of CNN models are trained to accurately classify different plant diseases, using models such as Xception, EfficientNet, MobileNet, and DenseNet.

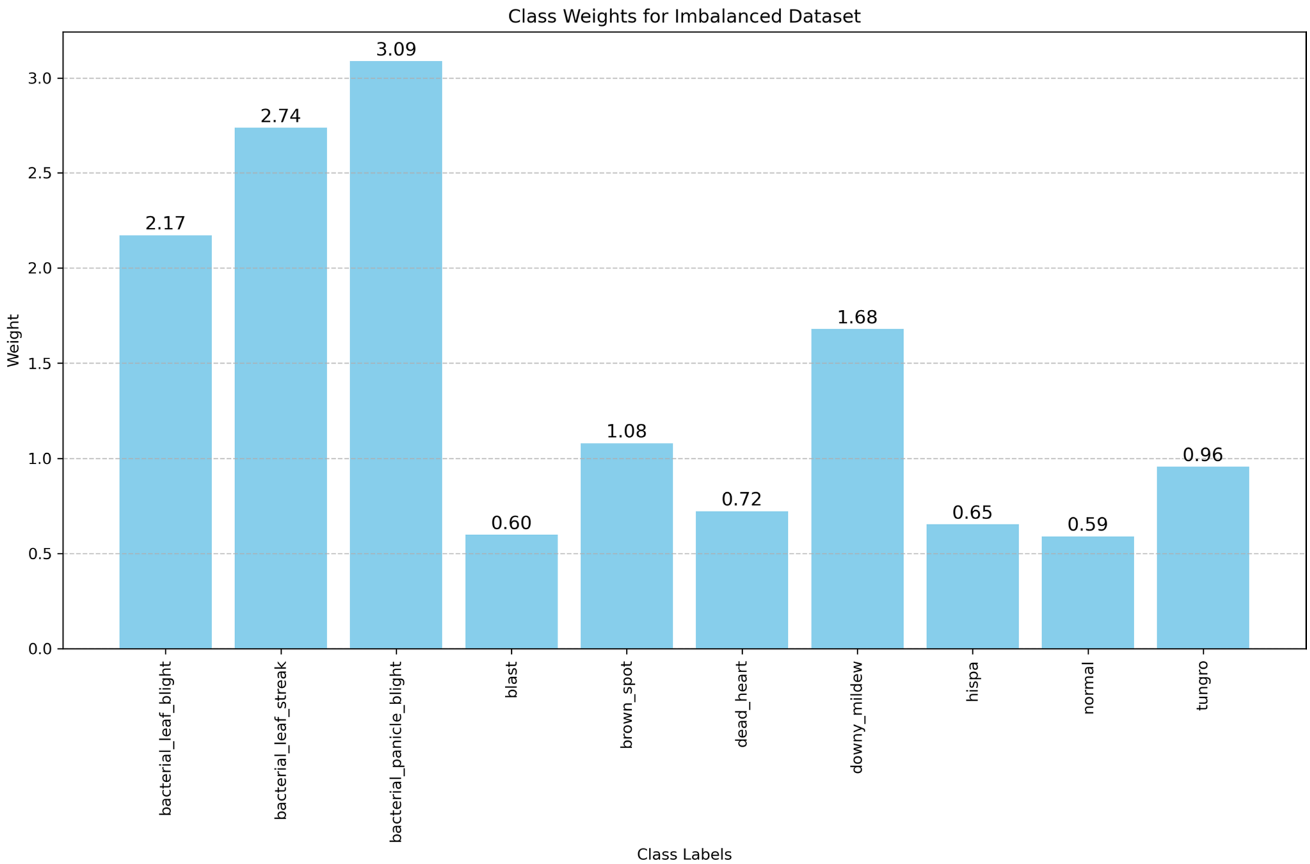


Fig. 2 Class weight graph

3.2.1 Xception algorithm

The Xception algorithm is a deep convolutional neural network architecture. The name "Xception" means "Extreme Inception," reflecting its foundation on the Inception architecture while substituting the Inception modules with depth wise separable convolutions. Xception's architecture comprises of a series of depth wise separable convolution blocks, where each block includes a depth wise convolution followed by a pointwise convolution. By decomposing the convolution into depth wise and pointwise operations, Xception reduces the number of parameters and computational complexity while maintaining high accuracy. The Xception architecture is a 71-layer architecture. These 71 layers can be categorized into three types: convolutional layers, fully connected layers, and depth wise separable convolutions. The fully connected layer has several neurons equal to the number of output classes [19].

3.2.2 MobileNet

MobileNetV3 Large is an efficient, resource-constrained model designed for devices such as smartphones. MobileNetV3 consists of a total of 53 layers. The output layer contains the same number of neurons as the number of classes in the dataset. It consists of four important components: the stem, inverted residual blocks, expansion layer, and head. The architecture is designed to decrease the number of parameters and computations while maintaining accuracy. It achieves this through the inverted residual block, which is more efficient than traditional residual blocks. The inverted residual block consists of depth wise convolution, pointwise convolution, and separable convolution, which reduce the parameter count and computation cost. MobileNetV3 achieved state-of-the-art performance in various image classification tasks [20].

3.2.3 DenseNet

DenseNet121 is a type of convolutional neural network from the DenseNet family, consisting of 121 layers. The DenseNet architecture features 10 different types of layers and is generally used in image classification, image segmentation, and object recognition. DenseNet121 represents the original DenseNet architecture, with other types being similar but with additional layers based on this architecture. It has approximately 7.9 million parameters and 1.3 million neurons. The computational cost is higher due to its architectural complexity. It is one of the models that has achieved state-of-the-art performance in several image classification and object detection tasks [21].

3.2.4 EfficientNet

EfficientNet-B4 is a type of convolutional neural network (CNN) which is part of the EfficientNet family. The "B4" refers to a specific configuration within this architecture. EfficientNet-B4 has a total of 55 layers, 5.3 million neurons, and 24.5 million parameters. It has achieved state-of-the-art results on various image tasks, demonstrating efficiency in image classification by minimizing the number of parameters and computational demands while preserving accuracy. However, it requires a large amount of data during training [22].

3.3 Training framework and tools

Lightning AI and WandB play crucial roles in managing and optimizing the training workflow in this study. Lightning AI provides a scalable and efficient infrastructure for training deep learning models on cloud-based GPUs, including NVIDIA T4 and L4. On the other hand, WandB served as a powerful experiment tracking and visualization tool, offering real-time insights into training metrics such as accuracy, loss, and resource usage. Its interactive dashboards allowed for dynamic monitoring of model performance. By integrating these tools, the study benefited from a streamlined workflow for handling large-scale training and analysis, ensuring optimal model performance and resource efficiency throughout the experimentation process.

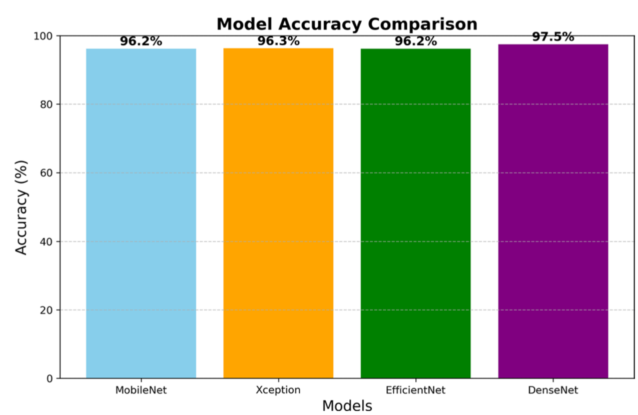
4 Results and discussion

4.1 Comparing test accuracy and test loss

Comparing Figs. 3 and 4, which show the accuracy and loss of the four models, DenseNet121 achieved the highest accuracy of 97.50% with a test loss of 13.46. The EfficientNetB4 model attained an accuracy of 96.25% and a loss of 23.66, while the MobileNetV3 Large model achieved the same accuracy of 96.25% but with a slightly lower loss of 21.99. The Xception model reached an accuracy of 96.32% and a loss of 17.29.

In the Fig. 3, DenseNet121 demonstrates the highest accuracy compared to the other models, with the remaining three models showing similar accuracy levels. In the Fig. 4, DenseNet121 has the lowest loss among the models, whereas EfficientNetB4 has the highest loss. MobileNetV3 Large has a lower loss than EfficientNetB4 but a higher loss compared

Fig. 3 Test Accuracy



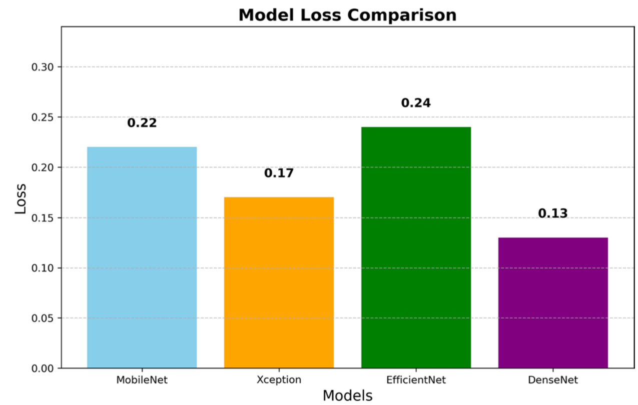


Fig. 4 Test Loss

to Xception. Overall, both figures indicate that DenseNet121 offers the best performance due to its superior accuracy (97.50%) and the lowest loss (13.46%).

4.2 Training loss

The training loss for all four models (Fig. 5) started above 2% in the first epoch and decreased significantly as training continued, though with different levels of stability. The MobileNet model (blue curve) started with a high loss (above 2.0), decreased quickly in the early steps, and smoothly reached 0.04% by the final epoch, showing effective learning.

Similarly, the Xception model (red curve) began with a loss above 2%, steadily decreased, and reached 0.05% by the final epoch. The DenseNet121 model (orange curve) also started with a loss above 2%, dropped quickly, but had noticeable fluctuations before settling at 0.14%.

The EfficientNetB4 model (green curve) showed a similar starting loss but had some instability with occasional spikes before stabilizing at 0.04% by the final epoch.

Among these models, EfficientNetB4 and Xception had the most stable loss curves, showing efficient and smooth convergence. Despite some fluctuations, all models achieved good optimization with significant reductions in training loss, indicating effective learning and performance.

4.3 Training accuracy

Fig. 6 shows the training accuracy plot, which reveals the learning progression of the four models. The MobileNet model (blue curve) steadily improves from an initial accuracy of around 0.6, quickly reaching 1.0 within the first few steps, and maintains high accuracy throughout training.

The DenseNet121 model (orange curve) follows a similar pattern, reaching near-perfect accuracy quickly, though it shows minor fluctuations later. The EfficientNetB4 model (green curve) starts at the same baseline, rapidly converges to 1.0, and maintains stable accuracy without major deviations, showing its efficiency in training.

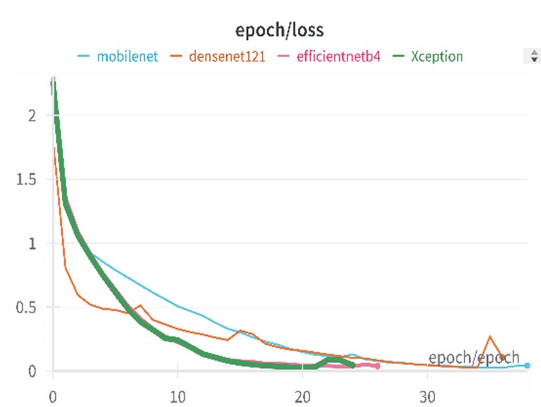
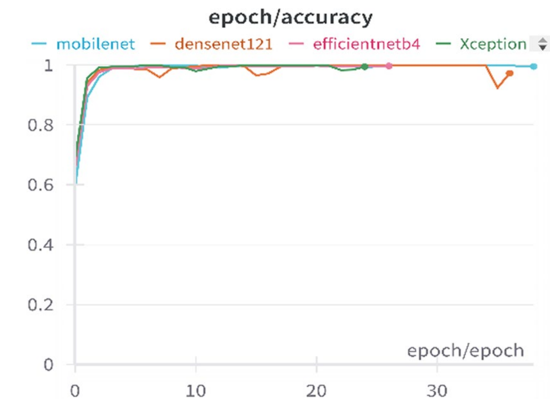


Fig. 5 Training Loss

Fig. 6 Training Accuracy



The Xception model (red curve) also reaches near-perfect accuracy early and remains consistent, similar to EfficientNetB4 in stability and performance.

Overall, all models show strong learning abilities, quickly reaching high accuracy. EfficientNetB4 and Xception are the most stable and reliable models in terms of training accuracy.

4.4 Validation accuracy

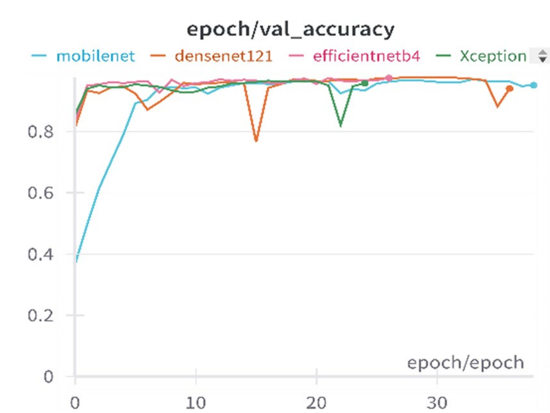
Fig. 7 shows the validation accuracy for all four models, which improved significantly during training, indicating effective generalization to unseen data. MobileNet (blue curve) started with 36.78% accuracy and steadily rose to 98.18%. Xception (red curve) began at 85.55% and reached 95.77%, with some fluctuations around step 25. DenseNet121 (orange curve) started at 81.38% and achieved 94.08%, with occasional drops. EfficientNetB4 (green curve) began at 83.33% and improved to 97.53%, maintaining stable performance.

EfficientNetB4 and Xception showed the most stable validation accuracy, while MobileNet steadily improved, and DenseNet121 performed well despite minor instability.

4.5 Validation loss

The validation loss for all four models (Fig. 8) demonstrated significant improvement throughout training, with varying levels of stability. MobileNet (blue curve) started with a high validation loss (>2.5), steadily declining and converging to approximately 0.5 by the final epoch, indicating improved generalization. Similarly, Xception (red curve) began at 2.8%, showing a consistent decline to around 0.4, though it experienced a temporary spike near step 20. DenseNet121 (orange curve) started with a lower initial validation loss of 1.7%, decreasing rapidly but showing fluctuations with occasional increases and decreases throughout training. EfficientNetB4 (green curve) also began at 1.7%, demonstrating an efficient and consistent drop to approximately 0.4, with minimal instability, reflecting its robustness and strong generalization capability. Among the models, EfficientNetB4 and Xception exhibited the most stable and efficient validation loss curves,

Fig. 7 Validation Accuracy



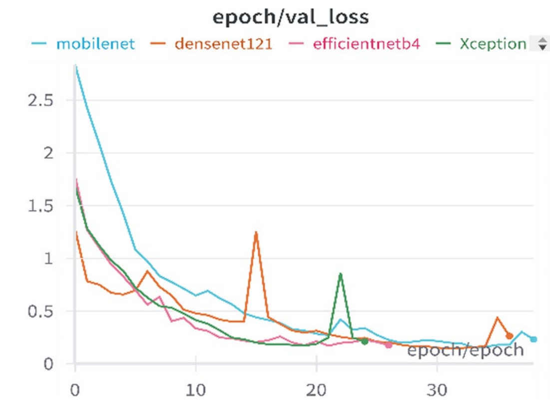


Fig. 8 Validation Loss

while MobileNet showed steady improvement with slightly slower convergence. DenseNet121 performed well overall but could benefit from tuning to mitigate fluctuations. All models made substantial progress in reducing validation loss, reflecting their learning and generalization capabilities.

Table2 shows the maximum GPU memory utilized, and RAM utilized during the training of each model. EfficientNetB4 and Xception had higher resource consumption with EfficientNetB4 requiring 5884 MB of RAM. DenseNet121 also demonstrated significant memory usage, but slightly less than EfficientNetB4, utilizing 2719 MB of RAM. In contrast, MobileNetV3 Large, designed for efficiency, was the most resource-efficient, using only 523 MB of RAM. These differences highlight the trade-offs between model complexity and computational efficiency, with MobileNetV3 being more suitable for resource-constrained environments.

4.6 Discussion

This study evaluates the performance of four state-of-the-art deep learning models-DenseNet121, EfficientNetB4, Xception, and MobileNetV3 Large-for rice plant disease detection. Among these, DenseNet121 achieved the highest accuracy of 97.5%, reflecting its robust feature extraction and classification capabilities. EfficientNetB4 and Xception also demonstrated strong performance, with accuracies of 96.25% and 96.32%, respectively. MobileNetV3 Large, while achieving a comparable accuracy of 96.25%, excelled in computational efficiency, making it ideal for resource-constrained environments.

4.6.1 DenseNet121 in real-time applications

Despite its superior accuracy, DenseNet121's suitability for real-time agricultural applications is limited due to its higher computational and memory requirements, as well as increased GPU utilization. These limitations restrict its deployment on mobile or edge devices, where lightweight and efficient models are preferred. However, DenseNet121 remains an excellent choice for cloud-based diagnostics, where computational resources are abundant.

4.6.2 Practical contributions to farmers

The findings hold significant implications for the agricultural sector. A mobile application leveraging MobileNetV3 for edge and mobile devices, or DenseNet121 for cloud-based diagnostics, could revolutionize paddy disease management. Such solutions could empower farmers with timely and accurate disease detection, reducing crop loss and improving

Table 2 Resource Utilization

Model	RAM Utilized (MB)	GPU Memory Utilized (%)	GPU Type
DenseNet121	2719	94.21	NVIDIA T4
EfficientNetB4	5884	93.98	NVIDIA T4
Xception	4787	93.80	NVIDIA T4
MobileNetV3 Large	523	93.68	NVIDIA L4

Table 3 Comparison with previous studies

Reference	Dataset	Size	Model	Performance
[13]	UCI Machine Learning Repository and Kaggle	984	VGG-19 ResNet-101 Xception Inception-ResNet-V2	VGG-19(81.43%) ResNet-101(91.52%) Xception(89.42%) Inception-ResNet-V2(92.68%)
[14]	Rice Leaf Disease dataset publicly available in Kaggle	115	CNNIR-OWELM	94.2%
[16]	Custom Dataset	2000	CNN	91.45%

yield quality. However, challenges such as connectivity issues, environmental variability, and infrastructure costs must be addressed for effective adoption.

Table 3 highlights the performance improvements in this study compared to previous works. The DenseNet121 model outperformed prior models, such as VGG-19, ResNet-101, Xception, and Inception-ResNet-V2, in terms of accuracy. The larger dataset used in this study (10,407 images) compared to datasets with 2,000 or fewer images in earlier studies significantly contributed to better generalization and performance. Additionally, the adoption of state-of-the-art architectures like DenseNet121 and MobileNetV3 Large marks a significant advancement over other models.

5 Conclusion

In this paper, we compare four state-of-the-art models that have demonstrated strong performance in various image classification tasks, applied to disease detection in rice plants. The four models are: DenseNet121, EfficientNetB4, Xception, and MobileNetV3 Large. Among these models, DenseNet121 achieved the highest accuracy, with a score of 97.5%. Both EfficientNetB4 and MobileNetV3 Large also performed well, each achieving an accuracy of 96.25%, while the Xception model demonstrated strong performance with an accuracy of 96.32%. The DenseNet121 model is recommended for applications where accuracy is the highest priority, whereas EfficientNetB4and MobileNetV3 Large are more suitable for deployment on devices with limited computational resources.

Future work will focus on expanding the dataset to include images captured under diverse environmental conditions, improving model generalization. Additionally, integrating multimodal approaches, such as com- bining environmental parameters with disease data, can enhance detection robustness. Linking the system with IoT devices for real-time monitoring and automated alerts will further improve practicality. Lastly, exploring advanced architectures like Vision Transformers (ViTs) and hybrid CNN-ViT models may lead to improved performance and scalability.

Author contributions A.B. conducted the experiments and provided the results for the manuscript. T.G. conceived the research idea and identified the methodologies used to conduct the experiments. M.K. wrote the main manuscript text, and A.K. prepared the figures. All authors reviewed the manuscript.

Funding The authors declare that no funds, grants, or other support were received during the preparation of this manuscript.

Data availability The data supporting the findings of this study are publicly available and have been properly cited in the article. Readers can access the data through the references provided in the manuscript.

Declarations

Competing interests The authors have no Competing interests to declare that are relevant to the content of this article.

References

1. Paddy P. Pandarasamy A (Samy): Paddy Doctor: Paddy Disease Classification. Kaggle https://kaggle.com/competitions/paddy-disease-classification (2022)
2. Ashwini S, Prashanthi SK, Krishnaraj PU, Iliger KS, Shruthi Mallikarjun K. Biological management of bacterial leaf blight of rice (xanthomonas oryzae pv. oryzae) through rice rhizosphere antagonist actinobacteria. Int J Environ Climate Change. 2023;13:660–73. https://doi.org/10.9734/ijecc/2023/v13i71918.
3. Liu Z, Friskop A, Jacobs JM, Dill-Macky R. Bacterial leaf streak: A persistent and increasingly important disease problem for cereal crops. Phytopathology. 2023;113:2020–3. https://doi.org/10.1094/PHYTO-11-23-0423-SA.
4. Azzahra RSN, Miranti M, Prismantoro D, Mohd Suhaimi NS, Alizadeh M, Mispan MS, Doni F. Bacterial panicle blight caused by burkholderia glumae: a major disease in rice cultivation. Cogent Food Agri. 2024. https://doi.org/10.1080/23311932.2024.2381600.
5. Jalalifar R, Sabouri A, Mousanejad S, Dadras AR. Estimation of genetic parameters and identification of leaf blast-resistant rice RILS using cluster analysis and MGIDI. Agronomy. 2023. https://doi.org/10.3390/agronomy13112730.
6. Dogra R, Rani S, Singh A, Albahar MA, Barrera AE, Alkhayyat A. Deep learning model for detection of brown spot rice leaf disease with smart agriculture. Comput Elect Eng. 2023. https://doi.org/10.1016/j.compeleceng.2023.108659.
7. Recent developments in plant-downy mildew interactions. Tör, M., Wood, T., Webb, A., Göl, D., M. McDowell. Sem Cell Dev Biol. 2023;148–149:42–50. https://doi.org/10.1016/j.semcdb.2023.01.010.
8. Sharma R, Kukreja V, Kadyan V. Hispa Rice Disease Classification using Convolutional Neural Network. Coimbatore India: IEEE; 2021.
9. Hore TK, Inabangan-asilo MA, Rea Wulandari. Introgression of tsv1 improves Tungro disease resistance of a rice variety BRRI dhan71. Sci Rep. 2022. https://doi.org/10.1038/s41598-022-23413-4.
10. Pacal I, Kunduracioglu I, Alma MH, Deveci M, Kadry S, Nedoma J, Slany V, Martinek R. A systematic review of deep learning techniques for plant diseases. Artif Intell Rev. 2024. https://doi.org/10.1007/s10462-024-10944-7.
11. Kunduracioglu PII. Advancements in deep learning for accurate classification of grape leaves and diagnosis of grape diseases. J Plant Dis Prot. 2024;131:1061–80. https://doi.org/10.1007/s41348-024-00896-z.
12. Aslan YE. Diagnosis and accurate classification of apple leaf diseases using vision transformers. Comput Dec Making: Int J. 2024;1:1–12. https://doi.org/10.59543/comdem.v1i.10039.
13. Islam MA, Shuvo MNR, Shamsojjaman M, Hasan S, Hossain MS, Khatun T. An automated convolutional neural network based approach for paddy leaf disease detection. Int J Adv Comput Sci Applic (2021) https://doi.org/10.14569/IJACSA.2021.0120134
14. Sowmyalakshmi R, Jayasankar T, Pilllai VA, Subramaniyan K, Pustokhina IV, Pustokhin DA, Shankar K. (2024) An optimal classification model for rice plant disease detection. Computers Materials Continua 68: 1751–1767 https://doi.org/10.32604/cmc.2021.016825
15. Li, C.S.M.H.e.a. R.: Predicting rice diseases using advanced technologies at different scales: present status and future perspectives. aBIOTECH 4, 359–371 (2023) https://doi.org/10.1007/s42994-023-00126-4
16. Haridasan A, Thomas J, Raj ED: Deep learning system for paddy plant disease detection and classification. Environ Monit Assess 2023 https://doi.org/10.1007/s10661-022-10656-x
17. Bakirarar B, ELHAN AH. Class weighting technique to deal with imbalanced class problem in machine learning Methodological research. Turkiye Klinikleri J Biostat. 2023;15:19–29. https://doi.org/10.5336/biostatic.2022-93961.
18. O'Shea, K., Nash, R.: An Introduction to Convolutional Neural Networks. 1511.08458 https://arxiv.org/abs/1511.08458 (2015)
19. Chollet, F.: Xception: Deep Learning with Depthwise Separable Convolutions. 1610.02357 https://arxiv.org/abs/1610.02357 (2017)
20. Qian S, Ning C, Hu Y. MobileNetV3 for Image Classification. Nanchang, China, 2021.
21. Huang G, Liu Z, Maaten LVD, Weinberger KQ. Densely Connected Convolutional Networks. 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 2017, 2261-2269 (2017)
22. Tan M, Le QV. EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks. Int Conf Machine Learn https://arxiv.org/abs/1905.11946 (2020)

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.