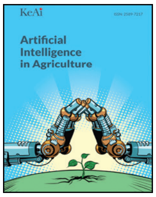
Contents lists available at ScienceDirect

Artificial Intelligence in Agriculture

journal homepage: http://www.keaipublishing.com/en/journals/artificial-intelligence-in-agriculture/



Comparison of CNN-based deep learning architectures for rice diseases classification

Md Taimur Ahad [a,*], Yan Li [b], Bo Song [c], Touhid Bhuiyan [d]

[a] Department of Computer Science and Engineering, Daffodil International University, Savar, Bangladesh
[b] School of Mathematics, Physics and Computing University of Southern Queensland, Toowoomba, Australia
[c] School of Engineering University of Southern Queensland, Toowoomba, Australia
[d] Department of Computer Science and Engineering, Faculty of Science and Information Technology, Daffodil International University, Bangladesh

ARTICLE INFO

Article history:
Received 10 June 2022
Received in revised form 6 July 2023
Accepted 7 July 2023
Available online 14 July 2023

Keywords:
Deep learning
Convolutional neural networks (CNNs)
Transfer learning
Plant leaf disease detection

ABSTRACT

Although convolutional neural network (CNN) paradigms have expanded to transfer learning and ensemble models from original individual CNN architectures, few studies have focused on the performance comparison of the applicability of these techniques in detecting and localizing rice diseases. Moreover, most CNN-based rice disease detection studies only considered a small number of diseases in their experiments. Both these shortcomings were addressed in this study. In this study, a rice disease classification comparison of six CNN-based deep-learning architectures (DenseNet121, Inceptionv3, MobileNetV2, resNext101, Resnet152V, and Seresnext101) was conducted using a database of nine of the most epidemic rice diseases in Bangladesh. In addition, we applied a transfer learning approach to DenseNet121, MobileNetV2, Resnet152V, Seresnext101, and an ensemble model called DEX (Densenet121, EfficientNetB7, and Xception) to compare the six individual CNN networks, transfer learning, and ensemble techniques. The results suggest that the ensemble framework provides the best accuracy of 98%, and transfer learning can increase the accuracy by 17% from the results obtained by Seresnext101 in detecting and localizing rice leaf diseases. The high accuracy in detecting and categorisation rice leaf diseases using CNN suggests that the deep CNN model is promising in the plant disease detection domain and can significantly impact the detection of diseases in real-time agricultural systems. This research is significant for farmers in rice-growing countries, as like many other plant diseases, rice diseases require timely and early identification of infected diseases and this research develops a rice leaf detection system based on CNN that is expected to help farmers to make fast decisions to protect their agricultural yields and quality.

© 2023 The Authors. Publishing services by Elsevier B.V. on behalf of KeAi Communications Co., Ltd. This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/).

1. Introduction

The observable success of convolutional neural networks (CNNs) has shifted the technology in detecting and localizing rice diseases using their leaves. Following the path, recent CNN studies suggest that the use of CNN has increased in rice leaf disease detection and segmentation (Chen et al., 2021; Patil and Kumar, 2022). Usually, a diseased rice leaf is covered with spots, colours, and diseased shapes (Mohanty et al., 2016). Thus, a diseased leaf which has a different colour texture and dimension than a healthy rice leaf provides an opportunity to perform image analysis using a CNN network and to collect information on inconsistency among the pixels of the entire leaf (Mitkal et al., 2016; Xu et al., 2020). Each of the pixels of a leaf is expected to be similar concerning any characteristic or available property, such as colour, intensity, or texture. However, in a case where a small group of pixels differs from others, it provides information about inconsistency in an object or the presence of other objects.

Despite this fact, a handful of research efforts were devoted (Chen et al., 2018; Akhter et al., 2019; Islam et al., 2018; Sarker et al. 2016) in detecting rice disease using CNN, however, there are still gaps in the CNN-based rice leaf disease detection research. Firstly, as rice diseases are different from country to country, a comprehensive study on most of the epidemic diseases in a given country should be conducted. The number of types of rice diseases should be increased instead of using a few key classes (Acharya et al., 2020). Secondly, a study should inform whether original CNN architectures, transfer learning, or ensemble techniques can provide better accuracy for rice leaf disease detection. Consequently, this study aims to examine the success of original state-of-the-art CNN architectures, such as DenseNet121 (Huang et al., 2017), Inceptionv3 (Adegun and Viriri, 2021), MobileNetV2 (Nagasubramanian et al., 2020), resNext101 (Adegun and Viriri, 2021),

[*] Corresponding author at: Department of Computer Science and Engineering, Daffodil International University, Savar, Bangladesh
E-mail addresses: MdTaimur.Ahad@usq.edu.au, taimurahad.cse@diu.edu.bd (M.T. Ahad).

https://doi.org/10.1016/j.aiia.2023.07.001

2589-7217/© 2023 The Authors. Publishing services by Elsevier B.V. on behalf of KeAi Communications Co., Ltd. This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/).

Resnet152V (Sankupellay and Konovalov, 2018) and Seresnext101 (Chen et al., 2018; He et al., 2015) in classifying the rice plant disease detection. The ultimate goal of this research is to reach the model that provides the highest accuracy rate in classifying rice leaf disease detection. Furthermore, this study also aims to conduct a transfer learning of DenseNet121, MobileNetV2, Resnet152V (Xie et al., 2017), and Seresnext101 (Chen et al., 2018) networks, and ensemble learning of Densenet121, EfficientNetB7, and Xception (Chollet, 2017) networks.

Another motivation to conduct this research is that rice disease detection is still preliminary among farmers in most developing countries with technical laggards. A commonly used method for rice disease detection is still manual visual inspection - simply using naked-eye observation by trained experts (Mohanty et al., 2016). In doing so, a large team of experts, as well as continuous monitoring, are required. This is costly and time-consuming for poor farmers when a farm is large. On the other hand, in some countries, farmers do not have proper mechanisms or even have no idea that they can consult with experts. Manual plant disease identification is a more laborious task, subject to human errors, and can be done only in limited areas. Therefore, automatic detection techniques have become increasingly in demand.

The main contributions of this paper are as follows: firstly, this research provides a comparison of six original CNN architectures, namely DenseNet121, Inceptionv3, MobileNetV2, resNext101, Resnet152V, and Seresnext101, using a dataset of the nine most epidemic rice diseases in Bangladesh. Secondly, we also applied a transfer learning approach on DenseNet121, MobileNetV2, Resnet152V, and Seresnext101 to conclude if transfer learning is capable of increasing accuracy, and thirdly, an ensembled model called DEX based on Densenet121, EfficientNetB7 and Xception networks were applied to draw a comparison among original, transfer learning, and ensemble techniques.

2. Literature review

This section reviews the related research work aiming at a deep-learning network model that can provide better accuracy for rice leaf disease detection. It includes an overview of the basic CNN networks and their various architectural structures. Several image processing technologies that were applied to rice leaf disease detection are also discussed.

2.1. Types of rice diseases

As a cereal grain, rice is the most widely consumed staple food for over half of the population in the world (Nguyen-Quoc and Hoang, 2020). Like many other developing countries, rice is the major source of income for rural farmers, especially in Bangladesh. Therefore, when rice production is hindered due to various rice diseases, it impacts the national economy.

Rice diseases are commonly involved in an abnormal physiological process that distorts the rice plant's normal structure, growth, and nutrition. Each country, however, has different rice diseases that impact people differently (Acharya et al., 2020). For example, rice yellow mottle disease is not known to occur in other parts of the world, except Tanzania (Huang et al., 2017). Blast rice disease causes 11–15% yield loss annually (Adegun and Viriri, 2021). Sarker et al. reported that Sheath blight, a rice disease caused by Rhizoctonia solani, affects the crop in almost every season in Bangladesh. The disease reduces the quality of the rice as well as the quantity of the crop, which in turn affects the economy of a country like Bangladesh where agriculture is the main sector (Nagasubramanian et al., 2020). However, among all the rice diseases, the following are the most adverse rice diseases in Bangladesh (Sankupellay and Konovalov, 2018):

1. **Bacterial Leaf Blight***: Xanthomonas Oryzae* Pvoryzae is responsible for bacterial leaf blight (BL), occurring mostly during the wet season, especially when water overflows through rice fields. In some areas in Asia, it can reduce crop yield by up to 50%, even up to 80% (Chen et al., 2018).
2. **Blight:** Bacterial panicle blight, also known as blight, is caused by gram-negative bacterial pathogens *Burkholderia glumae* and *B. gladioli*. It is a prevalent disease in many rice-growing regions globally. The symptoms of bacterial panicle blight include several characteristic features. Panicle discolouration is a common symptom, where affected panicles may turn dark brown or black. Another symptom is grain rot, where the infected grains may exhibit decay or rotting. Additionally, sterile florets, which fail to produce viable seeds, are observed in affected panicles (Sainath et al., 2015; Zhu and Gong, 2018).
3. **Brown Spot:** Brown leaf spots are another serious rice disease worldwide, caused by *Bipolaris Oryzae* (Breda de Haan) Shoemaker. The symptoms are leaf spots throughout the growing season, mostly on the leaf blade, small spots of dark brown to reddish-brown, circular to oval in shape, while older spots have a light, reddish-brown or grey centre surrounded by a dark to reddish-brown margin (Zhu and Gong, 2018).
4. **Hispa***:* Rice Hispa, scientifically known as *Dicladispa Armigera*, is a major pest that affects rice plants. It is commonly found during the tillering stage of rice growth, with higher populations observed at this stage. Rice Hispa feeds on the epidermis of the upper leaves, causing scraping damage. In the context of Sylhet, a region in Bangladesh, it is mentioned that Rice Hispa forms part of a large and contiguous population that extends into neighbouring Assam. This suggests that the pest is prevalent and poses a significant problem in both areas (He et al., 2015).
5. **Leaf Scald:** Leaf scald, caused by *Microdochium Oryzae*, is a fungal disease. This causes the scalded appearance of leaves. Zonate lesions of alternating light tan and dark brown starting from leaf tips or edges, oblong lesions with light brown halos in mature leaves. Individual lesions are 15 cm long and 0.51 cm wide or may almost cover an entire leaf. The continuous enlargement and coalescing of lesions result in the blighting of a large part of the leaf blade.
6. **Leaf Smut***:* Leaf smut, caused by the fungus Entyloma Oryzae, is a type of smut disease that affects rice plants. However, it does not produce smut balls on spikelets. Instead, leaf smut causes dark brown to black lesions on the leaves of the rice plant. These lesions are composed of masses of fungal spores. False smut causes chalkiness of grains which leads to a reduction in grain weight. It also reduces seed germination. Velvety smut balls on spikelets, Spore balls are initially orange and turn greenish black when mature, they are some identifiable symptoms of leaf smut. The second stage of infection occurs when the spikelet nearly reaches maturity.
7. **Leaf Blast:** Blast disease is a rice disease caused by Rhizoctonia solani. It affects the crop in almost every season in Bangladesh, causing 11–15% yield loss annually reported that Sheath blight. The disease reduces the rice quality as well as its quantity, which in turn affects the economy of a country like Bangladesh where agriculture is the main sector (Chen et al., 2018).
8. **Shath Blight:** Sheath blight is caused by the fungal pathogen Rhizoctonia solani. It is a widespread disease that affects rice plants in both temperate and tropical rice-growing regions. The typical symptoms of sheath blight manifest as oval to irregular lesions on the rice sheath (the protective covering of the stem) and leaf blades. These lesions typically have a greyish inner colour and a dark brown margin. The distinct colouration helps in identifying and distinguishing the disease from other rice pathogens.
9. **Tungro:** Tungro is a viral disease that affects rice plants. It is distributed in South and Southeast Asia, including Bangladesh. The disease is characterized by several characteristic features, such as stunted growth of the plant, twisted leaves, reduced tillering (formation of side shoots) and delayed flowering. The main vector responsible for the transmission of rice tungro virus is the green planthopper

(Nephotettix virus). In Bangladesh, susceptible rice varieties are particularly susceptible to Tungro disease. Studies have shown that tungro incidences were common in susceptible cultivars, with recorded incidence rates ranging from 85% to 81% in the south to northwest of the country. This indicates a significant impact of the disease on rice cultivation in these regions.

2.1. Convolutional neural networks

In CNNs, the term "convolutional" refers to the mathematical operation of convolution, which combines two functions to produce a third function. In the context of CNNs, convolutional layers apply filters or kernels to the input data, resulting in the generation of feature maps. These feature maps capture different aspects or patterns in the input data.

CNNs consist of multiple layers arranged sequentially. The basic structure typically includes an input layer, followed by several convolutional layers, pooling layers, a fully connected layer, and finally an output layer. The convolutional layers are responsible for extracting features from the input data through the application of filters. The pooling layers help reduce the spatial dimensions of the feature maps, reducing computational complexity. The fully connected layer connects the extracted features to the output layer, allowing the network to make predictions or perform classification based on the learned features.

The input and output layers are considered the visible layers of the CNN, while the intermediate layers, such as convolutional and pooling layers, are referred to as hidden layers. These hidden layers play a crucial role in learning and extracting hierarchical representations from the input data. Overall, the combination of convolutional layers, pooling layers, and fully connected layers in CNNs allows the network to effectively learn and recognize patterns in complex data, making them particularly well-suited for tasks such as image recognition and computer vision (Xie et al., 2017).

2.1.1. State-of-the-art CNN architectures. In this study, several CNN architectures are divided into three broad categories, original, transfer learning, and ensemble technique. The architecture of a basic CNN is given in Fig. 1.

2.1.2. Original CNN networks. An original CNN architecture in this research refers to a CNN network and algorithm that is available in Keras or Github. In this research, a CNN algorithm is kept original as first proposed by its authors and programmers, with no change of processing units, parameters and hyper-parameter optimization strategies, design patterns and connectivity of layers. Often a well-known CNN network was developed and evolved by different researchers and programmers through various challenges For example, the AlexNet architecture was the winner of ILSVRC 2012 and was proposed by Krizhevsky et al., (Chollet, 2017) ResNet was proposed by He et al. (Chollet, 2017) from Microsoft and won 2015 ILSVRC. DenseNet as an extension of ResNet was first proposed in 2016 by Huang et al. from Facebook (Chollet, 2017; Patil and Kumar, 2022). Several original CNN architectures with the versions are discussed in the next sections:

2.1.2.1. DenseNet-121. DenseNet-121 architecture iteratively concatenates the feature maps from one layer to another layer along the network, which is useful for classification tasks. The DenseNet-121 model was claimed better than MobileNetV2, ResNet50 and NASNet architectures (Shujaat et al., 2021).

2.1.2.2. Inception V3. Inception V3 developed by Google is the third release in the Deep Learning Evolutionary Architectures series. The Inception V3 architecture, which has the Softmax function in the last layer, consists of 42 layers in total and the input layer takes images with 299 × 299 pixels.

2.1.2.3. MobileNetV2. Overall, MobileNetV2's design choices make it well-suited for deployment on resource-constrained mobile devices while maintaining competitive performance for various computer vision tasks such as image classification, object detection, and semantic segmentation. One key feature of MobileNetV2 is the use of an inverted residual structure with residual connections between the bottleneck layers. Inverted residual blocks aim to reduce computational complexity while maintaining accuracy. The residual connections help in gradient flow and facilitate the training of deeper networks. The intermediate expansion layer in MobileNetV2 incorporates lightweight depth-wise convolutions (referred to as depth-wise separable convolutions) to introduce non-linearity and filter features efficiently. Depth-wise convolutions separate the spatial and channel-wise convolutions, reducing the computational cost while retaining the expressive power of the network.

The architecture of MobileNetV2 typically starts with an initial fully convolutional layer with 32 filters. This layer processes the input data

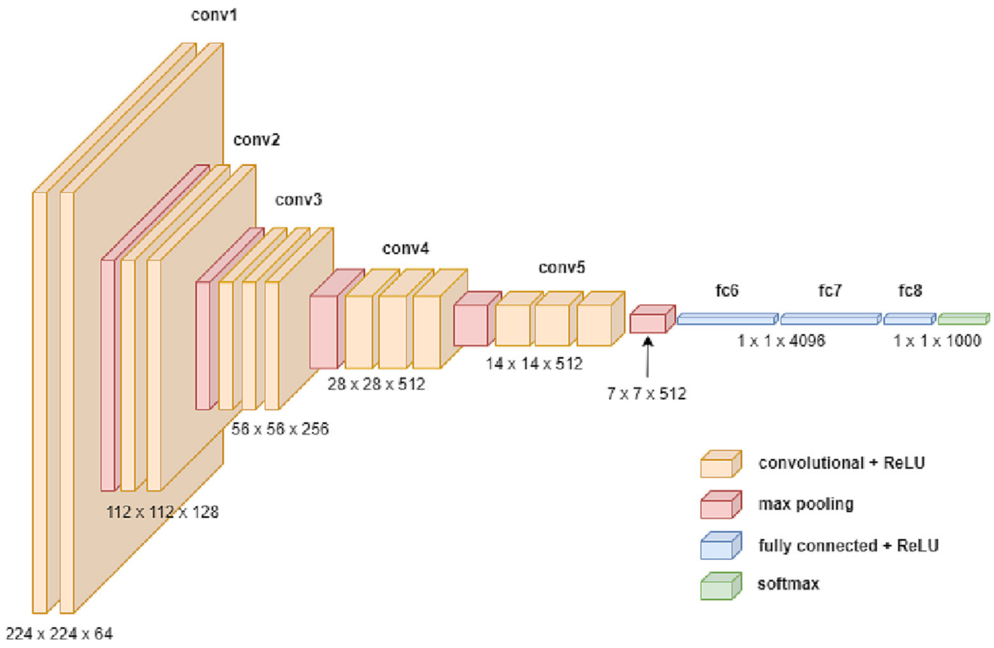


Fig. 1. The architecture of a basic CNN.

and extracts initial features. It is followed by 19 residual bottleneck layers, which are the primary building blocks of the network. These bottleneck layers further refine and transform the features, utilizing the inverted residual structure and depth-wise convolutions (Shujaat et al., 2021).

2.1.2.4. ResNeXt. ResNeXt, short for Aggregated Residual Transform Network, is a CNN architecture that builds upon the concepts of Residual Networks (ResNets) and Inception Networks. It introduces the idea of a split, transform, and merge block and emphasizes the concept of cardinality to improve performance. In ResNeXt, a split, transform, and merge block is used, where multiple transformations are applied within the block. These transformations help in learning diverse representations and capturing different levels of abstraction. The cardinality parameter is introduced to define the number of transformation paths within the block. Increasing cardinality has been shown to enhance the model's performance. ResNeXt has demonstrated impressive performance in various computer vision tasks, including image classification, object detection, and image segmentation. By combining the concepts of ResNets, Inception Networks, and cardinality-based transformations, ResNeXt achieves improved accuracy while maintaining computational efficiency. It's worth noting that the details of ResNeXt's architecture, such as the number of layers, cardinality, and block configurations, may vary depending on specific implementations and variations. Different versions of ResNeXt have been proposed to optimize the trade-off between performance and computational cost for different tasks and datasets (Patil and Kumar, 2022).

2.1.2.5. ResNet. ResNet, short for Residual Network, introduced the concept of residual layers and skip connections to address the issue of vanishing gradients in deep neural networks. The main innovation of ResNet is the residual layer, which allows the network to learn residual mappings rather than attempting to learn the full mapping from the input to the desired output directly. By using residual layers, the network can more easily capture and propagate gradients through the network, even in very deep architectures. To facilitate the flow of gradients and address the vanishing gradient problem, ResNet incorporates skip connections, also known as shortcut connections or identity mappings. These connections enable the gradients to flow directly from the end layers to earlier layers, bypassing the intermediate layers. This allows the gradients to propagate more effectively, preventing them from diminishing or vanishing as they backpropagate through the network. The skip connections in ResNet also enable the network to be significantly deeper than previous architectures. ResNet models have been successfully built with depths of 50, 101, or even 152 layers. The ability to train and effectively optimize such deep networks has been a major advancement in the field of deep learning. By using residual layers and skip connections, ResNet mitigates the vanishing gradient problem and facilitates the training of very deep neural networks. This has led to improved performance and accuracy in various computer vision tasks, including image classification, object detection, and image segmentation (Alegbejo et al., 2006; Ou, 1980).

2.1.3. Transfer learning. One approach, a transfer learning technique, is based on the knowledge gained from a training dataset and is used for training a different but relevant task or field (Weiss et al., 2016). In this deep learning process, the first few layers are trained to define the characteristics of the task. The last few layers of the trained network can be removed and retrained with new layers for the target task. It refers to the situation whereby what has been learned in one setting is adopted to improve the optimization in another setting. With limited computational requirements, ResNeXt-101 achieved state-of-the-art power and time speed (Albawi et al., 2017).

Training a deep learning model with a small dataset is often insufficient for its model's performance. Transfer learning is a process of pre-initialize a model using the weights obtained by training a different model on a larger, different, dataset. In the work conducted by Karimi et al. (2021), it was reported that although transfer learning reduced the training time on the target task, accuracy improvement depends on data quality. Large improvements are observed only when the segmentation task is more challenging and the target training data is smaller (Hossain et al., 2017).

2.1.4. Ensemble technique. Ensemble learning is one of the deep learning technologies that combine multiple primary learners through a fusion strategy to improve overall generalization performance (He et al., 2015). Ensemble learning has attracted a lot of attention because of its easy-to-understand structure and promising classification performance by combining more than one CNN model. Ensemble learning is a technique that incorporates multiple models for final decision-making The ultimate goal of an ensemble is that by combining multiple models, the errors of a single model can be corrected (compensated for) by other models, making the overall score (prediction and classification) of the ensemble better than any individual participating model (Kawasaki et al., 2015).

3. Literature review

The literature review suggests that four approaches can be used for the automatic diagnosis of rice diseases.

The first approach to automatic rice disease detection is through conventional means, such as pattern recognition techniques (Chen et al., 2018; Akhter et al., 2019). The study by Phadikar Sil (Islam et al., 2018) proposed an approach to rice disease identification in which the diseased rice images were classified using a self-organizing map (SOM) (via a neural network) that extracted the train images characteristics of the infected parts of a leaf were obtained from the rice diseases, while four different types of images were used for testing purposes. Satisfactory classification results have been reported. Sarker et al. (2016) presented a technique that uses only one feature, namely red, green, and blue (RGB) values, to detect and classify rice diseases. Using image processing, a disease was identified based on percentages of RGB values of an affected region. After extracting the RGB percentages from the affected region and grouping them into different classes, they were fed into a simple classifier called Naive Bayes, which classified diseases into different categories. Three rice diseases were successfully detected and identified using this technique: rice, brown spot, rice blight and rice blast. This technique was efficient and faster because only one feature (RGB values) extracted from the affected area was used, requiring minimal computation time to identify and classify the diseases. Instead of processing the whole leaf, this technique was successful in detecting the diseases using only small parts.

The second method is to use a Support Vector Machine (SVM). For example, Albawi et al. (2017); Hossain et al. (2017) used this method. Alfred et al. (2021) proposed an automated approach to classify rice plant diseases, brown spot diseases and leaf smut diseases based on their morphological changes. A total of 1000 spot images taken with a Nikon COOLPIX P4 digital camera of a rice field were used. Results were reported with 79.5% and 68.1% accuracy on the Bayesian and SVM classifiers, respectively. An SVM technique was also used by (Singh and Misra, 2017) for multiclass classification to identify three types of rice diseases (bacterial leaf blight, brown spot and leaf smut). The images of infected rice plants were taken with a digital camera from a paddy field and achieved an accuracy of 93.33% on the training data set and 73.33% on the test data set.

The third approach is the digital image processing techniques of McNeely-White et al. (2020); Atila et al. (2021); Chambon et al. (2021). Zhou et al. investigated a technique for assessing the extent of hop disease in rice crops, using a fuzzy C-means algorithm to classify regions into one of four classes: no disease, light disease, moderate disease, and severe disease. Their study achieved an accuracy of 87% in distinguishing cases in which a planthopper did or did not occur, while the accuracy in distinguishing four groups was 63.5%. Chambon et al. (2021) was to identify and classify six types of mineral deficiencies in rice. The study used features such as texture and colour for a developed specific multi-layer neural network. Both networks consist of a hidden layer with a different number (40 for texture and 70 forcolourr) of neurons in the hidden layer, in which 88.56% of the pixels were correctly classified. Similarly, the same authors proposed another similar work that successfully identified two types of diseases (blast and brown spot) affecting rice plants (Chambon et al., 2021).

The fourth approach is texture analysis and feature extraction using computer vision for enhancing the accuracy and rapidity of diagnosing the results. Phadikar and Sil (2008) developed an approach of texture analysis to identify four rice diseases (bacterial leaf blight, blast, brown spot and tungro virus) using fractal Fourier. In their proposed study, the image of a rice leaf was converted to CIELab colour space, and the system was able to achieve an accuracy of 92.5% (Phadikar and Sil, 2008). The features extracted from diseased and unaffected leaf images, the grey level co-occurrence matrix (GLCM) and the colour moment of the leaf lesion region were implemented by Lu et al. (2017) to create a 2-dimensional —D feature vector and related features. Redundant features were eliminated with a genetic algorithm-based feature selection method to generate 14-D feature vectors to minimize complexity. The technique has shown a promising result. However, to improve its detection accuracy, there is a need for more optimization procedures to take place. The key features of rice diseases, the brown spot and blast diseases, were described utilizing the colour of texture of rice leaf photos by Phadikar and Sil (2008). However, the efficiency of rice disease identification needs to be improved.

Phadikar et al. (2012), the entropy-based bipolar threshold technique was employed for the segmentation of the image after improving its brightness and contrast. The author sought to integrate image processing and soft computing technique for the detection of rice plants attacked by several types of diseases. The idea behind the technique was robust when utilized effectively. However, the average accuracy of identification on the four datasets was 82% which indicated that more enhancement was still required. Image processing and machine learning methods were utilized to non-destructively screen seedlings with rickets (Islam et al., 2018). Moreover, genetic algorithms were employed to develop SVM classifiers to optimize feature selection and model parameters for differentiating healthy seedlings and infected ones. The overall accuracy achieved in their study was 87.9%. Since a disease may have several different symptoms at the same time, this approach should be tested if other diseases are also present. It suggests that this approach has its limitations. Therefore, deep learning-based models became popular to detect diseases in various plants.

Singh and Singh (2010) study performed rice plant disease detection with a deep CNN. Using a VGGNet architecture, researchers at Chen et al. (2021) performed the classification of rice plant diseases. Chen et al. also proposed a CNN model, namely MobileNet Beta, by extending a pre-trained MobileNetV2 model to detect plant diseases (Chen et al., 2020). Too et al. reported that the DenseNet architecture achieved a high test accuracy of 99.75% (Praveen Kumar and Domnic, 2019). Geetharamani and Pandian trained a 9-layer CNN architecture using the PlantVillage dataset and achieved a classification accuracy of 96.46% using the test dataset (Too et al., 2019). Mohanty et al. (2016), on the other hand, used AlexNet and GoogLeNet to classify plant diseases and achieved a classification accuracy of 99.35% Geetharamani and Pandian (2019). Using the PlantVillage dataset, the Ferentinoss model of the VGG architecture delivered the highest accuracy at 99.53% (Arnal Barbedo, 2013).

Zhou et al. (2013) reported an automatic identification and diagnosis of rice diseases using CNN as a deep learning method. Using a dataset of 500 natural images of diseased and healthy rice leaves and stems captured from the rice experimental field, a CNN network was trained to identify 10 common rice diseases. Under the 10-fold cross-validation strategy, the proposed CNN-based model achieved an accuracy of 95.48%.

Sanyal and Patel (2008) suggested a faster R-CNN approach, which seemed to be ideal for the detection of rice diseases due to its good speed and high accuracy. Shrivastava et al. (2019) also applied a CNN algorithm for rice plant disease classification using a transfer learning of deep convolution neural network. Using an AlexNet CNN model, the model was able to classify rice diseases with a classification accuracy of 91.37%.

Asfarian et al. (2014) developed a CNN approach for detecting diseases and pests (five classes of diseases, three classes of pests, and one class of healthy plants and others) from rice plant images. A total number of 1426 images were collected that were captured using four different types of cameras and the system achieved a mean validation accuracy of 94.33%.

Akhter et al. (2019) also suggested a new stacked CNN architecture that used two-stage training to substantially reduce the model size while retaining a high classification accuracy. Several CNN architectures, such as MobileNet, NasNet Mobile, and SqueezeNet, were used. Experimental results showed that the proposed architecture achieved the desired accuracy of 93.3% with a significantly reduced model size, for example, 99% smaller than that of VGG16.

3.1.1. Knowledge gap in rice LEAF disease detection using CNN

Despite the fact, Phadikar et al. (2012) observed that computer-aided rice disease detection and classification have received special attention, Asfarian et al. (2014) criticized for low accuracy rates using the rice disease detection models. Our literature review in this study also suggest that the classification accuracies by most of the existing methods are between 50% and 95% (Asfarian et al., 2014). Moreover, those achieving higher accuracies were usually tested with fewer diseases. The performance would deteriorate if more diseases were included. (Acharya et al., 2020) and (Huang et al., 2017) discussed the gap between the current capabilities of image-based methods for automatic rice disease identification and the real-world implementation needs.

4. Methodology

The experiments in this study were conducted based on Google CoLab using the Keras library. TensorFlow which is one of the best Python deep learning libraries available for working with machine learning methods on Python was used. In this study, the original, transfer learning and ensemble models were trained using google collab Tesla graphics processing unit (GPU). TPU is available through the Google Collaboratory framework by Google. Initially, the colab framework provides up to 12 GB random access memory (RAM) and about 360 GB GPU in the cloud for research purposes.

4.1. Datasets

Data collection was the exceedingly cardinal quest for our research. We have put a vast effort to gather a great number of datasets. Since this research aimed to detect rice diseases, that mainly occurred in Bangladesh, most rice epidemic diseases found in the country were considered. Therefore, the data collected from rice leaf images included a combination of the Rice Leaf Disease Dataset from the University of California Irvine (UCI) Machine Learning Repository, a dataset from publicly available respiratory and a dataset collected from Bangladesh Rice Research Institute (BRRI). An example of rice leaves with various diseases is given in Fig. 2.

The final combined dataset contains nine (9) classes of rice diseases, with each class having one hundred (100) images for each type of disease. Images in the dataset are coloured images of various sizes and have a white background. The original images were divided into training and test sets with a ratio of 70:30 (see Table 1).

4.2. Process of experiments

The processes of the experiments are described in Fig. 3.

(1) Image Acquisition: In this step, we downloaded the images from the targeted sites to provide as input. Images in the dataset were checked manually to identify if they had a white background. In the case where images (mainly from the BRRI) had coloured backgrounds, images were placed on a white background. If disease symptoms such as spots, diseased colour, and diseased shape were not visible in an image, the image was removed from the dataset.

(2) Image Augmentation: Image augmentation is used in this step. Image augmentation is the procedure by which an existing dataset is expanded by transforming the original dataset to create more new data, and in such a way that new data are also label-preserving (Sankupellay and Konovalov, 2018, Meeras Salman Al-Shemarry et al., 2019). The goal is to increase the variance of the dataset while ensuring that new data are meaningful and do not merely add unnecessary volume to the dataset (Sankupellay and Konovalov, 2018). When used in a machine-learning context, it can improve model generalization, make trained models more robust to unseen data, and increase model accuracy (Sankupellay and Konovalov, 2018).

With these aims, we conducted data augmentation in the training data. However, position augmentation such as scaling, cropping, flipping, rotation, and colour augmentation such as brightness, contrast, and saturation was deployed. Random rotation from −15 degrees to 15 degrees, rotations of multiple of 90 degrees at random, random distortion, shear transformation, vertical flip, horizontal flip, skewing and intensity transformation were also used as part of the data augmentation process. In this way, 10 augmented images from every original image have been created. Random choice of a subset of the transformations helps augment an original image in a heterogeneous way.

In this study, each pixel value of images in the original and augmented images was first normalized dividing by 255. The images were then resized to a default size accepted by each model. In our experiment, input image resolutions were necessarily resized for all models of EfficientNet architecture due to our hardware limitations. Through trial and error, it was seen that the maximum allowed input size that our hardware resources were sufficient for the training of the EfficientNet model which has the highest number of parameters of 132 × 132. Therefore, the input size for all models of EfficientNet architecture was set as 132 × 132 to evaluate all models under the same conditions. Table II summarizes the default image resolutions and the number of parameters defined for deep learning models.

Table 1
Images used in the train, test and validation sets.

	Total images	Training images	Validation images
Original Dataset	900	630	270
Augmented Dataset	42,876	34,992	7884

(3) Training: A CNN learner model is created at this stage. By using DenseNet, EfficientNetB3, MobileNet, VGG16 and ResNet10 architectures, a model was trained based on the given dataset and then tested its classification accuracy.

All the models were trained for 175 epochs (iterations) with Early Stopping callbacks (patience = 10 iterations) Patience is the number of epochs with no improvement after which training will be stopped. An Adam optimizer, a combination of Stochastic Gradient Descent (SGD) with momentum and RMSProp (Root Mean Squared Propagation, or RMSProp, is an extension of gradient descent and the AdaGrad version of gradient descent that uses a decaying average of partial gradients in the adaptation of the step size for each parameter.) were used for faster convergence with the parameters like learning rate was set at $\alpha\alpha = 0.0001$, $\beta 1\beta 1 = 0.9$, $\beta 2\beta 2 = 0.999$ and $\epsilon = 1 \times 10 - 7\epsilon = 1 \times 10 - 7$. The same optimizer was used for all three models and then the models were saved as .h5 files. The time taken for model training is −31 s (s)/epoch (Iterations) for DenseNet201 and 17 s/epoch for each of the models ResNet50V2 and Inceptionv3.

In this research *standard deviation* was used as a model performance metric since the dataset used in this experiment does not have any major imbalance. *Categorical cross-entropy* was used as a loss function for all CNN architectures since this work deals with multi-class classification. All intermediate layers of the CNN architectures used in this work have *relu* as the activation function while the activation function used in the last layer was *softmax*. The hyperparameters used are as follows: the dropout rate was 0.3, the learning rate was 0.0001, the batch size was 64, and the number of epochs was 275. An adaptive moment estimation (Adam) optimizer was used for updating the model weights. All the images were resized to the default image size for each prior architecture.

(4) Classification: In this step, neural networks (*DenseNet121, Inceptionv3, MobileNetV2, resNext101, Resnet152V, resnext101, and Xception*) were used in the automatic detection of leaf diseases. The neural network was chosen as a classification tool due to its well-known technique as a successful classifier for many real applications. After the training model, the evaluation model was built for rice disease detection based on the highest probability of occurrence, the images of rice leaves were classified into different disease classes using a *softmax* output layer.

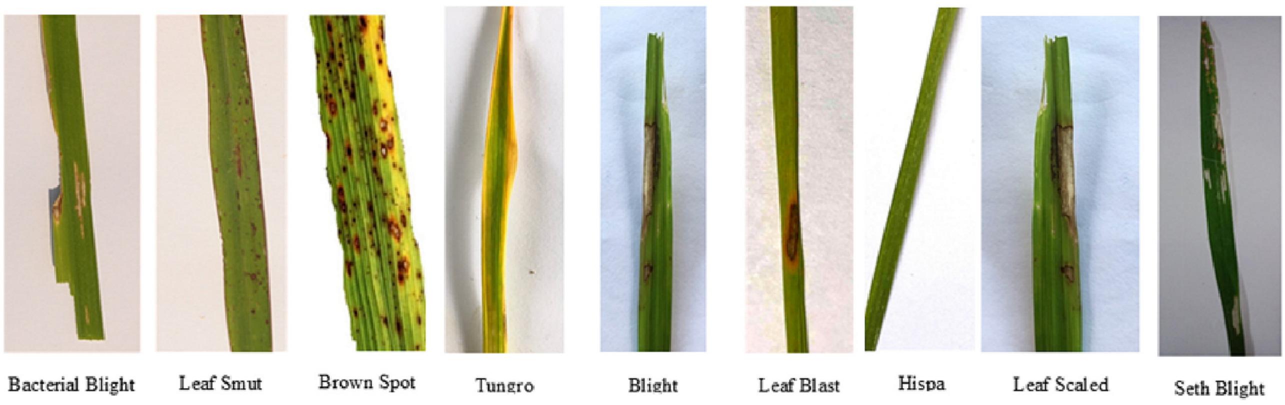


Fig. 2. Images examples of the rice diseases used.

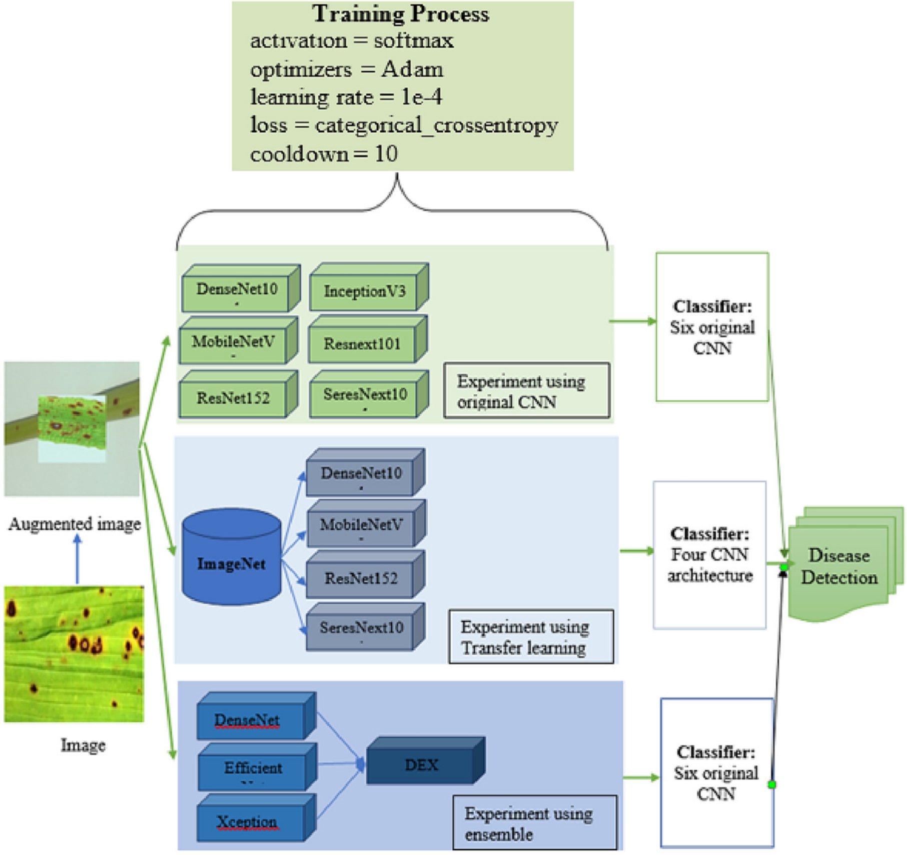


Fig. 3. Process of experiments.

5. Experimental results

The results from the experiments are presented in three sections based on the architectures of original individual networks, transfer learning and ensemble techniques. The results obtained are expected to answer the following questions:

1. Which original CNN network provides better accuracy in detecting rice leaf disease?
2. Does transfer learning improve accuracy?
3. Does the ensemble technique improve the accuracy?

Several performance measures for machine learning classification models are used to assess how well those CNN base algorithms perform in a given context. The following performance metrics are considered:

5.1. Accuracy

Accuracy is one metric for evaluating classification models. Informally, accuracy is the fraction of predictions our model got right. Formally, Accuracy is the ratio of correctly labelled images to the total number of samples (Kawasaki et al., 2015). The formula for accuracy is given below (1):

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \tag{1}$$

5.2. Precision

Precision is defined as the probability given a positive label, and how many of them are positive (Ferentinos, 2018). Precision tells us how many of the correctly predicted cases turned out to be positive. Precision is a useful metric in cases where FP is a higher concern than FN. The formula of precision is given below (2):

$$Precision = \frac{TP}{TP + FP} \tag{2}$$

5.3. Recall

Recall or Sensitivity is the accuracy of positively predicted instances describing how many were labelled correctly (Kawasaki et al., 2015). Recall tells us how many of the actual positive cases we were able to predict correctly with our model. The recall is a useful metric in cases where FN trumps FP. The formula of recall is given below (3):

$$F1 - score = \frac{2}{\frac{1}{Recall} + \frac{1}{Precision}} \tag{3}$$

F1-score, as an additional measure for classification accuracy, considers both precision and recall. F1-score is a harmonic mean of Precision and Recall, and so it gives a combined idea about these two metrics. It is maximum when Precision is equal to Recall.

5.4. Specificity

Specificity refers to the ability of a diagnostic test to correctly identify a rice leaf that is healthy or free from disease. It measures the percentage of true negative results. A highly specific test has a low false positive rate. However, in case of a highly specific test can be interpreted with confidence as a strong indication that the rice leaf is a diseased one. The equation is given below (4):

$$Specificity = \frac{TN}{TN + FP} \quad (4)$$

5.5. Training loss and validation loss

Training loss is a measure of how well a model fits the training data. It quantifies the discrepancy between the predicted output of the model and the actual target values in the training set. The goal during training is to minimize this loss, which indicates that the model is learning to accurately represent the relationship between the input data and the corresponding output targets.

Validation loss, on the other hand, assesses how well the model generalizes to new, unseen data. It measures the discrepancy between the model's predictions and the true target values in a validation set or a portion of the training data that is held out for evaluation. The validation loss helps determine if the model has learned meaningful patterns or if it is overfitting.

Overfitting occurs when a model becomes too complex or too closely fits the training data. In such cases, the model may start capturing noise or irrelevant patterns from the training set, making it less effective at generalizing to new data. Overfitting is often characterized by a low training loss but a high validation loss, indicating that the model is not performing well on unseen data.

To combat overfitting, techniques such as regularization, dropout, and early stopping can be employed. Regularization methods help prevent the model from excessively fitting the training data by introducing penalties or constraints on the model's parameters. Dropout randomly deactivates a portion of the neurons during training, reducing the model's reliance on specific features or patterns. Early stopping stops the training process when the validation loss starts to increase, preventing the model from further overfitting.

The aim is to strike a balance where the model minimizes both the training loss and the validation loss, indicating that it is learning meaningful patterns without overfitting the data. This ensures that the model generalizes well and performs accurately on unseen data.

5.6. Confusion matrix

A confusion matrix is a table that summarizes the results of a classification model by comparing the predicted labels with the true labels of a dataset. It provides a comprehensive view of the model's performance by displaying the counts of true positive (TP), false positive (FP), true negative (TN), and false negative (FN) predictions. Confusion matrices are valuable tools in evaluating and comparing different models, selecting appropriate thresholds, and understanding the trade-offs between various performance measures. They offer a clear and concise summary of the model's predictive performance and are widely used in machine learning and classification tasks.

5.7. Support

Support refers to the number of actual occurrences or instances of a particular class within a dataset. It represents the frequency or prevalence of a specific class. Imbalanced support occurs when there is a significant disparity in the number of instances between different classes in the training data. For example, if one class has a much larger number of instances compared to another class, the dataset is considered imbalanced. Imbalanced support can pose challenges in training classifiers and evaluating their performance. Classifiers tend to be biased towards the majority class due to the larger number of instances, resulting in lower accuracy or performance metrics for the minority class. This imbalance can indicate potential structural weaknesses in the reported scores of the classifier, as the overall performance may not accurately reflect its ability to correctly classify all classes. To address imbalanced support, various techniques can be employed. Stratified sampling is one approach that ensures each class is represented proportionally in the training and evaluation datasets. This helps provide a more balanced representation of classes during model training and evaluation. Rebalancing techniques, such as oversampling the minority class or undersampling the majority class, can also be used to mitigate the effects of imbalanced support during training.

5.8. Which original CNN network provides better accuracy in detecting rice LEAF disease?

In this section, the performances of the six original individual CNN networks (DenseNet121, Inceptionv3, MobileNetV2, resNext101, Resnet152V, and Seresnext101) are presented. The classification performance of the models is first presented. The overall measures for those models are then discussed. Gathering in addition to the descriptors, possible causes, and areas of opportunity for improvement of results.

Table 2 displays the accuracy of the DenseNet121, Inceptionv3, MobileNetV2, resNext101, Resnet152V, Seresnext101. The DenseNet121 and Inceptionv3 models achieved the highest accuracy at 97% and Seresnext101 gave the lowest accuracy value of 79%.

The Precision, Recall, F1-score and Specificity obtained by the DenseNet121, Inceptionv3, MobileNetV2, resNext101, Resnet152V, and Seresnext101 models for each of the classes are shown in Table 3. Considering the precision values for each on the test dataset, DenseNet121, Inceptionv3, and MobileNetV2, architectures provide the best performance. The above table suggests that the DenseNet121, Inceptionv3, and MobileNetV2 models classified Blight, Leaf Blast and Tungro diseases with 99% accuracy. The Seresnext101 performed low precision having the lowest identification of the Bacterial blight leaf with only 56% accuracy. All models identified Hispa with an average accuracy by the six models. Seresnext-101 requires large amounts of images for training to learn accurate representations compared with DenseNet121, Inceptionv3, MobileNetV2, resNext101, and Resnet152V. If the network receives less testing data, it provides lower classification accuracy. Moreover, the architecture and hyperparameters of the SEResNeXt-101 model could impact its performance. If the model architecture is not suitable for the specific image classification task, or if hyperparameters such as learning rate, batch size, or regularization settings are not properly tuned, it could result in lower accuracy. Lastly, since the SEResNeXt-101 model is too complex and has too many parameters relative to the size of the training dataset, leading to overfitting and reducing performance on new images (See Table 4).

5.9. Does transfer learning improve accuracy?

In this section, the performance of four transfer learning CNN architectures is presented. Table 6 shows the accuracies obtained in the test sets by DenseNet121, Seresnext101, EfficientNet and, Xception models. The test accuracies shown in Table 5 were calculated as the ratio of the number of correctly classified samples to the number of all samples. The DenseNet1121 model achieved the highest accuracy of 97%. However, the accuracy improvement from the original network to transfer learning by the SeresNext101 network is mentionable. The network

Table 2
Classification accuracy of the individual CNN networks in detecting rice diseases.

Architecture	Accuracy
DenseNet121	97%
Inceptionv3	97%
MobileNetV2	94%
resNext101	96%
Resnet152V	93%
Seresnext101	79%

Table 3
Precision, Recall, f1 and Specificity result of CNN networks with transfer learning.

	Bacterial leaf blight	Blight	Brown Spot	Hispa	Leaf blast	Leaf scaled	Leaf smut	Sheath Blight	Tungro
Densenet121									
Precision	96%	99%	97%	97%	99%	96%	96%	94%	99%
Recall	100%	100%	92%	99%	93%	97%	97%	95%	99%
F1-score	98%	100%	95%	98%	96%	96%	97%	95%	99%
Support (N)	872	864	871	870	869	854	869	870	869
Specificity	99.97%	100.00%	99.07%	99.87%	99.11%	99.67%	99.67%	99.52%	99.90%
Inceptionv3									
Precision	96%	99%	97%	97%	99%	96%	96%	94%	99%
Recall	99%	100%	92%	97%	90%	99%	100%	96%	97%
F1-score	98%	100%	95%	94%	95%	97%	98%	98%	97%
Support (N)	868	869	864	871	867	869	867	867	866
Specificity	100.00%	99.99%	98.48%	99.81%	98.78%	99.74%	99.09%	99.93%	99.97%
Mobilenetv2									
Precision	96%	99%	97%	97%	99%	96%	96%	94%	99%
Recall	99%	100%	92%	97%	90%	99%	100%	96%	97%
F1-score	98%	100%	95%	94%	95%	97%	98%	98%	97%
Support (N)	868	869	864	871	867	869	867	867	866
Specificity	99.93%	99.99%	99.07%	99.97%	98.81%	99.94%	99.94%	99.57%	99.69%
Resnet152v2									
Precision	94%	100%	100%	84%	86%	89%	96%	97%	98%
Recall	99%	100%	89%	87%	82%	96%	99%	95%	94%
F1-score	96%	100%	94%	85%	84%	93%	97%	96%	96%
Support (N)	872	869	864	866	868	870	870	863	866
Specificity	99.91%	100.00%	98.78%	98.44%	97.97%	99.51%	99.85%	99.41%	99.27%
Resnext101									
Precision	96%	100%	93%	97%	99%	96%	96%	94%	99%
Recall	100%	100%	92%	99%	93%	97%	97%	95%	99%
F1-score	98%	100%	95%	98%	96%	96%	97%	95%	99%
Support (N)	872	864	871	870	869	854	869	870	869
Specificity	99.24%	99.72%	96.39%	95.15%	98.34%	97.72%	99.66%	99.25%	99.40%
Seresnext101									
Precision	56%	87%	79%	84%	72%	96%	80%	95%	81%
Recall	93%	86%	40%	71%	65%	73%	89%	95%	99%
F1-score	70%	87%	53%	77%	69%	83%	84%	95%	89%
Support (N)	867	860	865	872	868	870	867	867	872
Specificity	99.26%	98.63%	94.23%	97.06%	96.51%	97.32%	98.84%	99.45%	99.93%

improved by a 17% accuracy increase after applying the transfer learning approach.

Table 6 shows the Precision, Recall, F1-score, and Specificity results from the CNN networks with transfer learning. In general, high Precision, high Recall, and high Specificity represent a better model. The experimental results demonstrate that SeresNext-101 had a low precision in detecting Bacterial leaf Blight with 56% accuracy. However, after transfer learning, the model reached 98% accuracy.

The associated TP, FN, FP, and TN are shown in Table 8. For rice disease detection and classification, we applied the seresNext101 Model with a transfer learning approach as the model received the lowest accuracy in earlier experiments (Without transfer learning). In addition to the SeresNext101 model, we also selected DenseNet121, EfficientNet and Xception models for rice leaf disease detection and classification. As these are deep convolutional networks and we were interested to see if the models are useful for small-scale datasets. The confusion matrix of DenseNet121, Seresnext101, EfficientNet and Xception is shown in Fig. 4.

5.10. Does the ensemble technique improve the accuracy?

In this research, the ensemble stack is developed on three different original CNN models, Densenet121, EfficientNetB7, and XceptionNet. To accelerate the training process, we adopted a transfer learning strategy. In addition to this, the output from these models was sent to a post-processing block containing a fully connected layer followed by a dropout layer and a final logit layer for classifying the image. For better convergence of our models, we used a learning rate decaying strategy which divided the learning rate by 10 only when the loss stops decreasing for three continuous epochs and an early-stopping strategy that halts the training process after the learning rate decayed 5 times (Kawasaki et al., 2015).

Table 4
Table 4 shows the values of TP, TN, FP, and FN by the different CNN architectures.

	Bacterial leaf blight	Blight	Brown Spot	Hispa	Leaf blast	Leaf scaled	Leaf smut	Sheath Blight	Tungro
Densenet121									
TP	870	864	804	861	805	830	845	836	862
FN	39	6	24	30	7	38	31	50	8
FP	2	0	67	9	64	24	24	34	7
TN	7130	7171	7146	7141	7165	7149	7141	7121	7164
Inceptionv3									
TP	867	863	756	857	778	852	805	860	863
FN	36	0	58	73	19	7	78	22	14
FP	0	1	111	14	89	19	66	5	2
TN	7212	7251	7190	7171	7229	7237	7166	7228	7236
Mobilenetv2									
TP	863	868	794	849	781	861	863	836	844
FN	37		9	82	3	54	29	8	27
FP	5	1	67	2	86	4	4	31	22
TN	7125	761	7160	7097	7160	7111	7134	7155	7137
resnet152v2									
TP	865	869	772	750	716	834	859	819	811
FN	60	4	2	148	118	99	34	26	16
FP	7		92	116	152	36	11	44	55
TN	7383	7442	7449	7301	7329	7346	7411	7426	7433
Resnext101									
TP	809	844	559	453	733	678	844	810	821
FN	78	72	70	74	418	59	255	89	105
FP	62	23	303	412	131	189	27	61	49
TN	8079	8089	8096	8089	7746	8102	7902	8068	8053
Seresnext101									
TP	808	742	345	616	566	635	769	820	866
FN	647	101	90	113	218	24	195	46	200
FP	59	118	520	256	302	235	98	47	6
TN	7928	8481	8487	8457	8356	8548	8380	8529	8370

Table 5
Accuracy results of CNN networks with transfer learning.

Architecture	Accuracy
DenseNet121	97%
Seresnext101	96%
EfficientNet	95%
Xception	92%

All the models have been trained for 60 epochs with Early Stopping callbacks (patience = 10 epochs). Adam optimizer, a combination of SGD with momentum and RMSProp, was used for faster convergence with the parameters as learning rate $\alpha\alpha = 0.0001$, $\beta 1\beta 1 = 0.9$, $\beta 2\beta 2 = 0.999$ and $\epsilon = 1 \times 10 - 7\epsilon = 1 \times 10 - 7$. The same optimizer is used for all three models and then the models are saved as .h5 files. The time taken for model training is −31 s/epoch for DenseNet201 and 17 s/epoch for each of the models ResNet50V2 and Inceptionv3. In Figure the gradual change in the loss function (both training as well as validation) through the epochs has been depicted for all three models of DenseNet201, ResNet50V2 and Inceptionv3. With a 97.62% (see Table 7) accuracy, the ensemble model outperformed the original CNN architecture (Densenet121, EfficientNetB7 and XceptionNet).

The precision on ensembling suggests that the model received 99% on Bacterial blight, which was 98% with transfer learning and 56% on the original CNN model (see Table 8). Even though the F1-score had the lowest accuracy (53% in the case of Brown Spot using Seresnext101) the ensemble model had 95% in that case). However, the Precision, Recall, f1 and Specificity result of CNN networks with the ensemble is shown in Table 8.

Fig. 5 shows the confusion matrix of the ensemble model. Fig. 6 shows the training accuracy and validation accuracy of the ensemble model of Densenet121, EfficientNetB7 and XceptionNet, where the x-axis represents the number of epochs and the y-axis represents the accuracy and loss percentages. Fig. 6 indicates that the training and validation data are split appropriately with no over-fitting.

Fig. 7 shows the training loss and validation loss over epochs by the ensemble technique. A loss function is used in CNN to optimize an architecture. The loss is calculated on training and validation and its interpretation is based on how well the model is doing in these two sets. It is the sum of errors made for each example in training or validation sets. Loss value implies how poorly or well a model behaves after each iteration of optimization. Fig. 7 suggests that the training loss was around 3% while the validation loss was 5% in 175 epochs.

Fig. 8 presents the accuracy of six different CNN-based models (DenseNet121, Inceptionv3, MobileNetV2, resNext101, Resnet152V and Seresnext101), transfer learning and ensemble model on nine classes of rice:

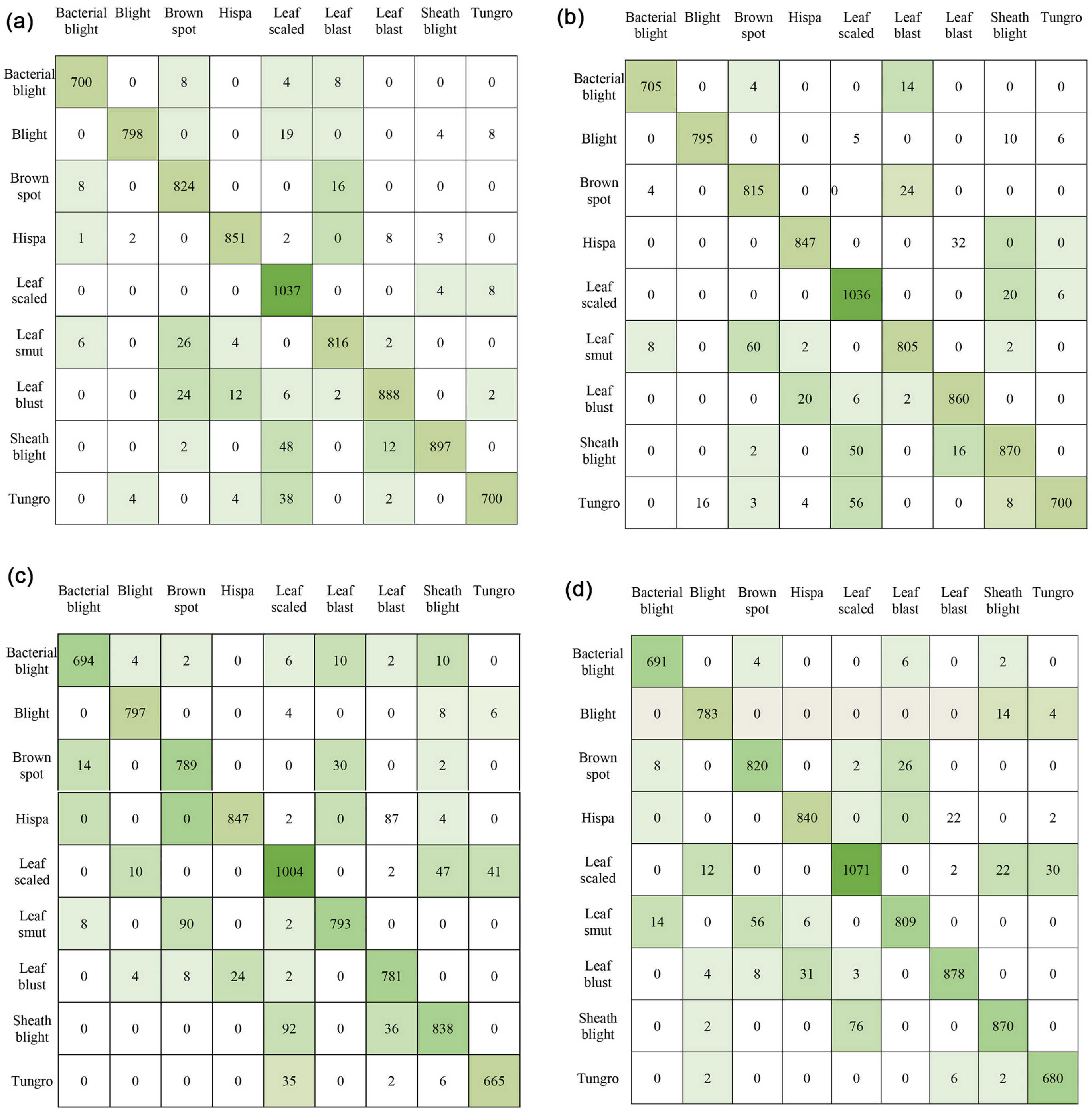
6. Discussions

In this research, we performed an in-depth investigation of the performances of original individual CNN, transfer learning, and ensemble models. We compared the results of six different CNN-based models of DenseNet121, Inceptionv3, MobileNetV2, resNext101, Resnet152V and SeresNext101 by applying them to the nine classes of rice diseases (see Fig. 8 for accuracy). The dataset used includes 14,118 rice leaf images. After image expansion through rotation, we obtained 34,992 images for training and 7884 images for testing. Among the original individual networks, Densenet121 provides the best classification results in identifying rice leaf diseases. Bari et al., 2021; Nayak and Singh, 2021 also support the findings that Denesenet121 delivers relatively high accuracy. This is because, in DenseNet, each layer obtains a “collective knowledge” from all preceding layers as layers receive inputs from all preceding layers and pass them on to the next layers.

Our investigation suggests that transfer learning of deep learning models provides slightly improved accuracy than the original individual networks for small datasets (Number of imageless than 2000). In this case, only after careful training including transfer learning, the accuracy was higher than the original CNN architecture. The transfer learning strategies in this research were based on using the pre-trained model for training and extracting features. Surprisingly we found that seresNext101 has improved by 17% of accuracy after a transfer learning process. This is consistent with the results from the study conducted by Oloko-Oba and Viriri (2021) that SE-ResNeXt-101 normally would involve more parameters and was computationally expensive but has shown good results on the ImageNet classification tasks. Performing transfer learning from images trained on Imagenet (general images such as cats, dogs, etc.) or MURA (X-ray images on different parts of the body but not the chest) improved results compared to scenarios when transfer learning was not used at all.

Table 6
Precision, Recall, f1 and Specificity result of CNN networks (Based on the number of images).

	Bacterial leaf blight	Blight	Brown Spot	Hispa	Leaf blast	Leaf scaled	Leaf smut	Sheath Blight	Tungro
Densenet121									
Precision	97%	96%	97%	98%	95%	99%	96%	94%	94%
Recall	98%	100%	93%	98%	97%	90%	97%	99%	97%
F1-score	98%	98%	95%	98%	96%	94%	96%	96%	95%
Support (N)	715	804	884	871	912	1154	842	908	718
Specificity	99.80%	99.92%	99.17%	99.72%	98.34%	99.64%	99.67%	99.85%	99.76%
Serenext101									
Precision	98%	96%	95%	97%	95%	94%	91%	92%	98%
Recall	98%	100%	92%	96%	97%	92%	91%	97%	95%
F1-score	98%	98%	94%	96%	96%	94%	94%	93%	97%
Support (N)	713	804	888	877	908	1152	841	910	716
Specificity	99.71%	99.73%	99.07%	99.49%	98.85%	99.64%	99.59%	99.45%	99.52%
EfficientNet									
Precision	98%	97%	97%	96%	97%	98%	92%	93%	90%
Recall	98%	98%	92%	97%	95%	90%	95%	96%	98%
F1-score	98%	98%	94%	97%	98%	94%	93%	94%	94%
Support (N)	717	811	883	869	908	1153	845	910	712
Specificity	99.84%	99.78%	99.07%	99.70%	98.36%	99.45%	99.34%	99.45%	99.84%
Xception									
Precision	95%	98%	94%	90%	95%	91%	89%	87%	94%
Recall	97%	98%	89%	97%	86%	88%	95%	92%	93%
F1-score	96%	98%	92%	94%	90%	90%	92%	90%	93%
Support (N)	716	815	889	871	910	1147	833	915	712
Specificity	99.73%	99.78%	98.84%	99.70%	98.21%	99.50%	98.38%	99.02%	99.42%



(a)

	Bacterial blight	Blight	Brown spot	Hispa	Leaf scaled	Leaf blast	Leaf blast	Sheath blight	Tungro
Bacterial blight	700	0	8	0	4	8	0	0	0
Blight	0	798	0	0	19	0	0	4	8
Brown spot	8	0	824	0	0	16	0	0	0
Hispa	1	2	0	851	2	0	8	3	0
Leaf scaled	0	0	0	0	1037	0	0	4	8
Leaf smut	6	0	26	4	0	816	2	0	0
Leaf blust	0	0	24	12	6	2	888	0	2
Sheath blight	0	0	2	0	48	0	12	897	0
Tungro	0	4	0	4	38	0	2	0	700

(b)

	Bacterial blight	Blight	Brown spot	Hispa	Leaf scaled	Leaf blast	Leaf blast	Sheath blight	Tungro
Bacterial blight	705	0	4	0	0	14	0	0	0
Blight	0	795	0	0	5	0	0	10	6
Brown spot	4	0	815	0	0	24	0	0	0
Hispa	0	0	0	847	0	0	32	0	0
Leaf scaled	0	0	0	0	1036	0	0	20	6
Leaf smut	8	0	60	2	0	805	0	2	0
Leaf blust	0	0	0	20	6	2	860	0	0
Sheath blight	0	0	2	0	50	0	16	870	0
Tungro	0	16	3	4	56	0	0	8	700

(c)

	Bacterial blight	Blight	Brown spot	Hispa	Leaf scaled	Leaf blast	Leaf blast	Sheath blight	Tungro
Bacterial blight	694	4	2	0	6	10	2	10	0
Blight	0	797	0	0	4	0	0	8	6
Brown spot	14	0	789	0	0	30	0	2	0
Hispa	0	0	0	847	2	0	87	4	0
Leaf scaled	0	10	0	0	1004	0	2	47	41
Leaf smut	8	0	90	0	2	793	0	0	0
Leaf blust	0	4	8	24	2	0	781	0	0
Sheath blight	0	0	0	0	92	0	36	838	0
Tungro	0	0	0	0	35	0	2	6	665

(d)

	Bacterial blight	Blight	Brown spot	Hispa	Leaf scaled	Leaf blast	Leaf blast	Sheath blight	Tungro
Bacterial blight	691	0	4	0	0	6	0	2	0
Blight	0	783	0	0	0	0	0	14	4
Brown spot	8	0	820	0	2	26	0	0	0
Hispa	0	0	0	840	0	0	22	0	2
Leaf scaled	0	12	0	0	1071	0	2	22	30
Leaf smut	14	0	56	6	0	809	0	0	0
Leaf blust	0	4	8	31	3	0	878	0	0
Sheath blight	0	2	0	0	76	0	0	870	0
Tungro	0	2	0	0	0	0	6	2	680

Fig. 4. (A): CM after TL of DenseNet121. (B): CM after TL of EfficientNetB7.(C): CM after TL of Xception. (D): CM after TL of Seresnext101.

Not surprisingly, from our investigation, we found that the ensemble of deep learning models improved its accuracy over a single CNN architecture. Our findings also support the study by (Acharya et al., 2020).

Table 7
Accuracy results of the Ensemble model.

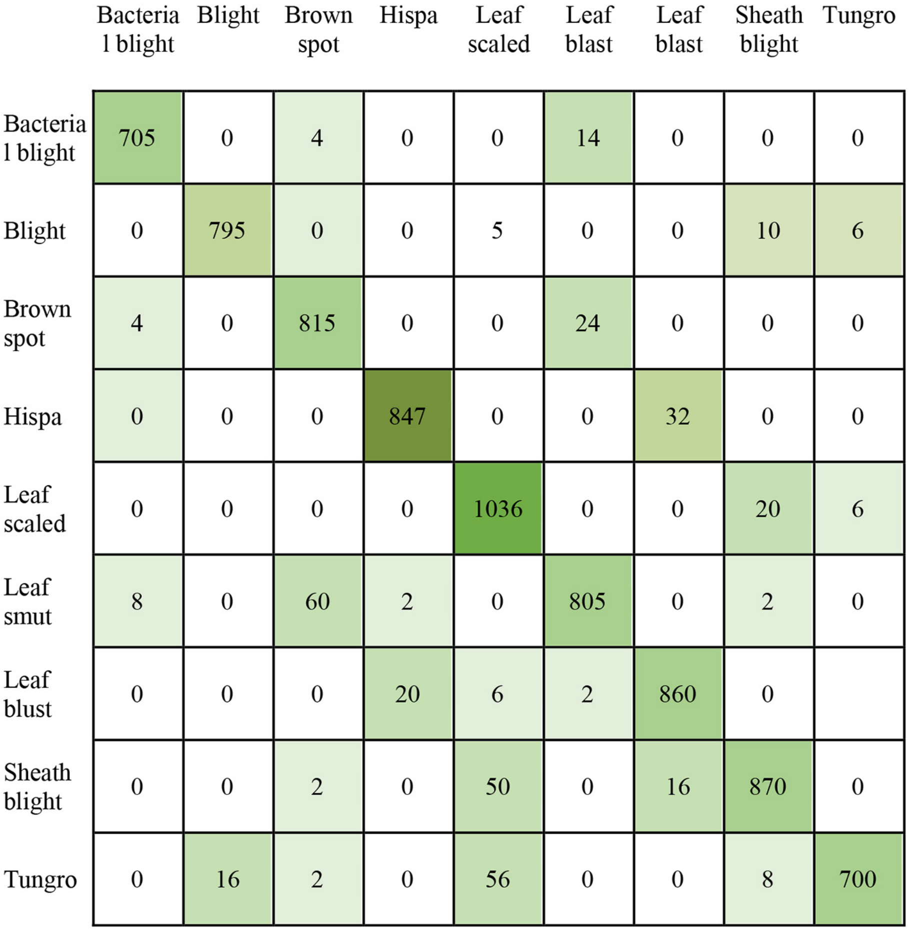
Architecture	Accuracy
Ensemble model (DEX)	97.62%

7. Contributions of this research

This research offers several key contributions. Firstly, this research experimented using nine types of rice diseases. Secondly, in this research, a comparison of six original CNN architectures (DenseNet121, Inceptionv3, MobileNetV2, resNext101, Resnet152V, and Seresnext101) was conducted. Thirdly, we applied a transfer learning approach on DenseNet121, MobileNetV2, Resnet152V, Seresnext101, and an ensemble model called DEX (Densenet121, EfficientNetB7, and Xception) to draw a comparison among the original CNN networks, transfer learning,

Table 8
Precision, Recall, f1 and Specificity result of ensembled CNN networks.

Ensemble DEX model (Densenet121, EfficientNetB7 & XceptionNet)									
	Bacterial blight	Blight	Brown Spot	Hispa	Leaf blast	Leaf scaled	Leaf smut	Sheath Blight	Tungro
Precision	99%	99%	99%	98%	97%	99%	94%	96%	96%
Recall	99%	100%	93%	99%	98%	94%	99%	99%	99%
F1-score	99%	99%	95%	99%	97%	97%	96%	96%	98%
Specificity	99.84%	100%	99.12%	99.70%	98.36%	99.45%	99.34%	99.45%	99.84%
Support	711	803	886	878	912	1146	844	913	715



	Bacteria l blight	Blight	Brown spot	Hispa	Leaf scaled	Leaf blast	Leaf blast	Sheath blight	Tungro
Bacteria l blight	705	0	4	0	0	14	0	0	0
Blight	0	795	0	0	5	0	0	10	6
Brown spot	4	0	815	0	0	24	0	0	0
Hispa	0	0	0	847	0	0	32	0	0
Leaf scaled	0	0	0	0	1036	0	0	20	6
Leaf smut	8	0	60	2	0	805	0	2	0
Leaf blust	0	0	0	20	6	2	860	0	
Sheath blight	0	0	2	0	50	0	16	870	0
Tungro	0	16	2	0	56	0	0	8	700

Fig. 5. CM of ensembled three CNN.

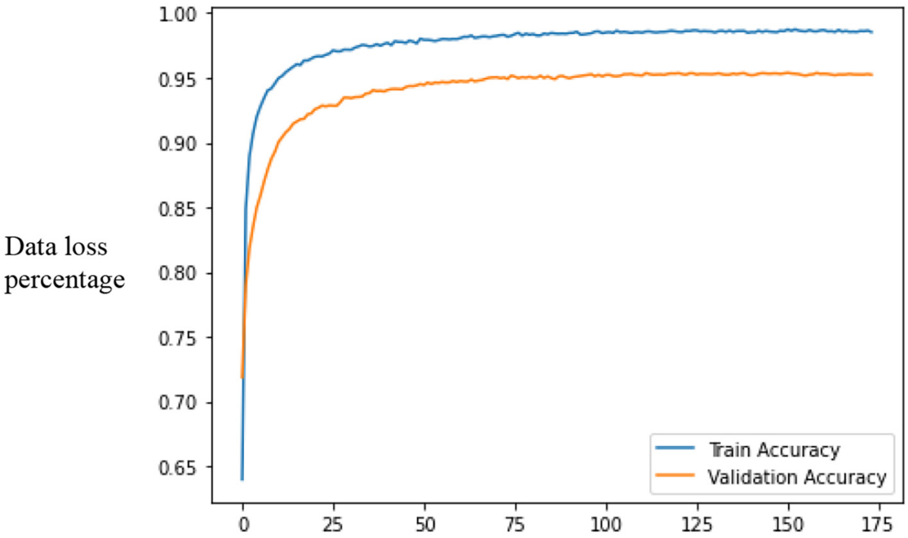


Fig. 6. Training and validation errors over the iteration of ensembled three CNN.

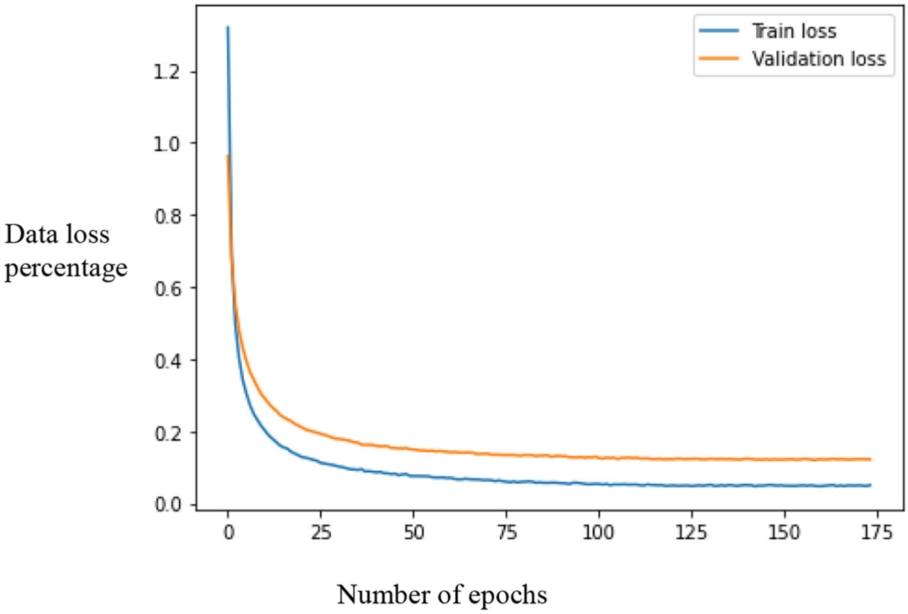


Fig. 7. Training and validation accuracy over the epochs of ensembled three CNN.

and ensemble technique. The results suggest that the ensemble framework provides the best accuracy of 98%, and the transfer learning increases a 17% accuracy from the results by Seresnext101 in detecting and localizing rice leaf diseases.

8. Conclusion and future research

There are some limitations in the current stage of the research, which need to address in future work. The use of free-of-charge resources (Google Colab) limits the experiments of this study. As Google Colab offers the server for a limited time, the hyperparameter tuning, training the base model training other than Imagenet (this research used Imagenet as the base database), and the application of Adadelta, FTRL, NAdam, Adadelta, and many more optimizers were not performed in this study. Another limitation is that the research used secondary data that are available publicly, not primary data directly collected from fields.

In the future, we want to create a user interface for the detection and localization of rice leaf diseases for farmers. This interface would not only detect but also provide a guide on how the diseases can be controlled. As mobile phones are seen as a preferred technological device among developing country users, we aim to develop a mobile phone-based rice leaf disease detection application tool.

The experimentation and the observations presented here are very important when models are being constructed with small datasets. In this research, the accuracy of the ensemble DEX model from

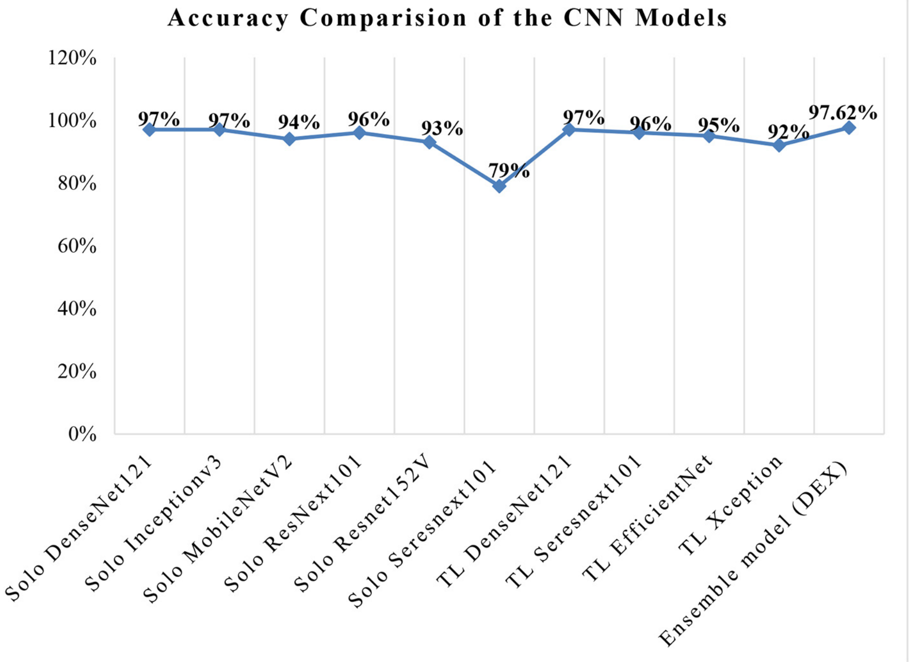


Fig. 8. Accuracy comparison among individual CNN, transfer learning and ensemble.

Densenet121, EfficientNetB7 & XceptionNet was found to produce the highest accuracy in classifying rice diseases from rice leaves. The success of the proposed architecture was compared with the transfer learning and six state-of-the-art individual CNN architectures. Experimental studies were conducted in both original and augmented versions of the image dataset. Considering both the average accuracy and the average precision metric on both the original and augmented datasets, the DEX model was found to be superior to other CNN architectures.

Funding

The work was not supported by any external funding.

CRediT authorship contribution statement

Md Taimur Ahad: Writing – original draft, Conceptualization, Methodology. **Yan Li:** Data curation, Writing – original draft, Supervision. **Bo Song:** Visualization, Investigation. **Touhid Bhuiyan:** Writing – review & editing.

Data availability

The data used to support the findings of this study are available from the corresponding author upon request.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

Acharya, A., et al., 2020, November. Plant disease detection for paddy crop using ensemble of CNNs. 2020 IEEE International Conference for Innovation in Technology (INOCON). IEEE, pp. 1–6.

Adegun, A., Viriri, S., 2021. Deep learning techniques for skin lesion analysis and melanoma cancer detection: a survey of the state-of-the-art. Artif. Intell. Rev. 54 (2), 811–841.

Akhter, M.S., et al., 2019. Plant virus diseases and their management in Bangladesh. Crop Prot. 118, 57–65.

Albawi, S., Mohammed, T.A., Al-Zawi, S., 2017. August). Understanding of a convolutional neural network. *2017 international conference on engineering and technology (ICET)*. Ieee, pp. 1–6.

Alegbejo, M.D., et al., 2006. Rice yellow mottle virus disease, a new disease of rice in Zamfara, Nigeria. Intern. Rice Res. Notes 31 (39), 39.

Alfred, R., et al., 2021. Towards paddy rice smart farming: a review on big data, machine learning, and rice production tasks. IEEE Access 9, 50358–50380.

Al-Shemarry, Meeras Salman, Li, Yan, Abdulla, Shahab, Wen, Peng, 2019. An efficient texture descriptor for the detection of license plates from vehicle images in difficult conditions. IEEE Transactions on Intelligent Transportation Systems. Vol 99, pp. 1–12. https://doi.org/10.1109/TITS.2019.2897990.

Arnal Barbedo, J.G., 2013. Digital image processing techniques for detecting, quantifying and classifying plant diseases. SpringerPlus 2 (1), 1–12.

Asfarian, A., et al., 2014. A computer vision for rice disease identification to support integrated pest management. Crop Prot. 61, 103–104.

Atila, Ü., et al., 2021. Plant leaf disease classification using EfficientNet deep learning model. Ecol. Inform. 61, 101182.

Bari, B.S., et al., 2021. A real-time approach of diagnosing rice leaf disease using deep learning-based faster R-CNN framework. PeerJ Comp. Sci. 7, e432.

Chambon, S., et al., 2021. When High-Performing Models Behave Poorly in Practice: Periodic Sampling Can Help.

Chen, C.F., et al., 2018. Big-little net: an efficient multi-scale feature representation for visual and speech recognition. arXiv preprint arXiv:1807.03848.

Chen, J., et al., 2020. Identifying plant diseases using deep transfer learning and enhanced lightweight network. Multimed. Tools Appl. 79 (41), 31497–31515.

Chen, J., et al., 2021. Identification of rice plant diseases using lightweight attention networks. Expert Syst. Appl. 169, 114514.

Chollet, F., 2017. Xception: deep learning with depthwise separable convolutions. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1251–1258.

Ferentinos, K.P., 2018. Deep learning models for plant disease detection and diagnosis. Comput. Electron. Agric. 145, 311–318.

Geetharamani, G., Pandian, A., 2019. Identification of plant leaf diseases using a nine-layer deep convolutional neural network. Comput. Electr. Eng. 76, 323–338.

He, K., et al., 2015. Delving deep into rectifiers: surpassing human-level performance on imagenet classification. Proceedings of the IEEE International Conference on Computer Vision, pp. 1026–1034.

Hossain, M., et al., 2017. Occurrence of blast disease in rice in Bangladesh. Am. J. Agricult. Sci. 4 (4), 74–80.

Huang, G., et al., 2017. Densely connected convolutional networks. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 4700–4708.

Islam, T., et al., 2018, April. A faster technique on rice disease detectionusing image processing of affected area in agro-field. 2018 Second International Conference on Inventive Communication and Computational Technologies (ICICCT). IEEE, pp. 62–66.

Karimi, D., Warfield, S.K., Gholipour, A., 2021. Transfer learning in medical image segmentation: New insights from analysis of the dynamics of model parameters and learned representations. Artificial intelligence in medicine 116, 102078.

Kawasaki, Y., et al., 2015, December. Basic study of automated diagnosis of viral plant diseases using convolutional neural networks. International Symposium on Visual Computing. Springer, Cham, pp. 638–645.

Kumar, J.P., Domnic, S., 2019. Image based leaf segmentation and counting in rosette plants. Inform. Proc. Agricult. 6 (2), 233–246.

Lu, Y., et al., 2017. Identification of rice diseases using deep convolutional neural networks. Neurocomputing 267, 378–384.

McNeely-White, D., et al., 2020. Inception and ResNet features are (almost) equivalent. Cogn. Syst. Res. 59, 312–318.

Mitkal, P., et al., 2016. Leaf disease detection and prevention using image processing using MATLAB. Intern. J. Recent Trends Eng. Res. (IJRTER) 2 pp.2455–1457.

Mohanty, S.P., et al., 2016. Using deep learning for image-based plant disease detection. Front. Plant Sci. 7, 1419.

Nagasubramanian, K., et al., 2020. Usefulness of interpretability methods to explain deep learning based plant stress phenotyping. arXiv preprint arXiv:2007.05729.

Nayak, J.K., Singh, P., 2021. Fundamentals of Research Methodology Problems and Prospects. SSDN Publishers & Distributors.

Nguyen-Quoc, H., Hoang, V.T., 2020. Rice seed image classification based on HOG descriptor with missing values imputation. TELKOMNIKA (Telecommunication Computing Electronics and Control) 18 (4), 1897–1903.

Ou, S.H., 1980. Pathogen variability and host resistance in rice blast disease. Annu. Rev. Phytopathol. 18 (1), 167–187.

Patil, R.R., Kumar, S., 2022. Rice-fusion: a multimodality data fusion framework for Rice disease diagnosis. IEEE Access 10, 5207–5222.

Phadikar, S., Sil, J., 2008, December. Rice disease identification using pattern recognition techniques. 2008 11th International Conference on Computer and Information Technology. IEEE, pp. 420–423.

Phadikar, S., Sil, J., Das, A.K., 2012. Classification of rice leaf diseases based on morphological changes. International Journal of Information and Electronics Engineering 2 (3), 460–463.

Sainath, T.N., Kingsbury, B., Saon, G., Soltau, H., Mohamed, A.R., Dahl, G., Ramabhadran, B., 2015. Deep convolutional neural networks for large-scale speech tasks. Neural Netw. 64, 39–48.

Sankupellay, M., Konovalov, D., 2018, November. Bird call recognition using deep convolutional neural network, ResNet-50. Proceedings of ACOUSTICS. Vol. 7, No. 9, pp. 1–8.

Sanyal, P., Patel, S.C., 2008. Pattern recognition method to detect two diseases in rice plants. Imag. Sci. J. 56 (6), 319–325.

Sarker, M.M., et al., 2016. Status of rice false smut disease in Natore district of Bangladesh. Bangl. Rice J. 20 (2), 31–37.

Shrivastava, P., Soon, T.K., Idris, M.Y.I.B., Mekhilef, S., 2019. Overview of model-based online state-of-charge estimation using Kalman filter family for lithium-ion batteries. Renewable and Sustainable Energy Reviews 113, 109233.

Shujaat, M., et al., 2021. Cr-prom: a convolutional neural network-based model for the prediction of rice promoters. IEEE Access 9, 81485–81491.

Singh, K.K., Singh, A., 2010. A study of image segmentation algorithms for different types of images. Intern. J. Comp. Sci. Iss. (IJCSI) 7 (5), 414.

Singh, V., Misra, A.K., 2017. Detection of plant leaf diseases using image segmentation and soft computing techniques. Inform. Proc. Agricult. 4 (1), 41–49.

Too, E.C., et al., 2019. A comparative study of fine-tuning deep learning models for plant disease identification. Comput. Electron. Agric. 161, 272–279.

Weiss, K., Khoshgoftaar, T.M., Wang, D., 2016. A survey of transfer learning. Journal of Big data 3 (1), 1–40.

Xie, S., et al., 2017. Aggregated residual transformations for deep neural networks. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1492–1500.

Xu, Zhe, Guo, Xi, Zhu, Anfan, He, Xiaolin, Zhao, Xiaomin, Han, Yi, Subedi, Roshan, 2020. Using deep convolutional neural networks for image-based diagnosis of nutrient deficiencies in rice. Comput. Intell. Neurosci. 2020.

Zhou, Z., et al., 2013. Rice plant-hopper infestation detection and classification algorithms based on fractal dimension values and fuzzy C-means. Math. Comput. Model. 58 (3–4), 701–709.

Zhu, X., Gong, S., 2018. Knowledge distillation by on-the-fly native ensemble. Adv. Neural Inf. Proces. Syst. 31.