University of Thi-Qar Journal
ISSN (print): 2706- 6908, ISSN (online): 2706-6894
Vol.20 No.2 June 2025

Performance Analysis Disease Detection Using Image Processing for Feature Extraction of Machine Learning Algorithms on Rice Leaf

Wasan M. Jwaid [1]*, Jehan Kadhim Al-Safi [2*], , Wijdan Rashid Abdulhussien [3*]

1 Department of Banking and Finance ,Administration and Economics, University of Thi-Qar, Nasiriyah, Iraq.

wasan.maktoof@utq.edu.iq

2 Department of Digital Media, Faculty of Media, University of Thi-Qar, Nasiriyah, Iraq.

jihan.k.shareef@utq.edu.iq

2 Department Information Technology, Computer Science and mathematics college, University of Thi-Qar, Nasiriyah, Iraq.
Wijdan_rashid@utq.edu.iq

https://doi.org/10.32792/utq/utj/vol20/2/5

Abstract

— Rice is the staple food for a large part of the world's population with such diseases interfering with the production of the staple food like bacterial blight, brown spot and smut disease among others. Identification of these diseases at an early stage and correct diagnosis can easily reduce the loss in yield and quality. Conventional approaches used in the identification of diseases in rice crops are executed through manual assessment which takes considerable time to complete hence compromising the identification of the diseases early enough to enable treatment prior to fruition of the disease's aunts. This has highlighted the need to implement an automated, efficient, and accurate means of diagnosing rice leaf diseases at their early stages.

The major goal of this project was the creation of a DL model that effectively identifies images of rice leaf diseases allowing for fast and efficient disease diagnostics and control. CNN was applied on a dataset of images of rice leaf diseases, each labeled in the three categories. The approach included data preparation (resizing the image, normalizing, augmenting), training the model using various architectures (Normal CNN, CNN with Data Augmentation, CNN with Hyperparameters tuning), and assessing the model using

correct and incorrect metrics. The results performing Normal CNN model pegged an accuracy level of 90% higher than CNN models with data augmentation and hyperparameter tuning levels of 85% and 75% respectively.
In conclusion, based on the evaluation of all models on the validation set, it was concluded that the Normal CNN model is the most suitable for this application in terms of its accuracy and the ability to generalize to new images that have not been used during the training process. This paper demonstrates how machine learning can be applied in the process of disease management in agriculture. The developed model not only improves the diagnosis speed and accuracy but also provides a new direction to diagnose other diseases in crops, creating great prospects for the development of agricultural practices and food security around the world.

Keywords: **Image feature-extraction; CNN; Rice plant disease;CNN-Data Augmentation.**

INTRODUCTION

Rice is the most widely consumed grain globally and is also known as the 'stick to the backbone' or 'white gold' [1]. Rice is among the most important food commodities that over 50% of the global human population count on for food and most especially in Asia where it is seen as the staple food and economic worth. Nevertheless, the production of this important crop is often threatened by adversities such as bacterial blight, the brown spot, and the smut of the leaf. Such diseases not only have an impact on productivity on crops but on food security of millions of farmers' across the globe. According to [2]–[4] the threats to Food Security: The diseases that affect rice crops are a major concern due to their potential consequences in food security as they cut down the annual yield by approximately 37 %. Complexity of Diagnosis Methods: In its current state, diagnosis of rice plant diseases is a complex, imprecise process that is restricted to specific regions. in addition, importance of Timely Diagnosis: Dissection on rice diseases important to reduce loss and determined that early diagnosis is crucial to prevent scaling [5]. Moreover, the Proposed Deep Learning Techniques: A new idea of using automated deep learning models has been put forward particularly to CNNs for diseases such as COVID-19, which has claimed traditional approaches [4], [6], [7]. It is an immense challenge to discuss rice diseases and its effects on the rate of agricultural productivity. Some diseases such as bacterial blight which affects the leaves, can lead to serious decrepitation and can reduce yield by up to 50 percent in the worst scenario. Dark spot, another common disease prevalent especially at the post-flowering stage, attacks the leaves hence affecting its ability to photosynthesis and thus reducing its grain quality as well as grain yield. Likewise, leaf smut in which dark, powdery-like spore masses develop on the surface of the leaves also affects the plant growth, fertility and leads to appreciable amount of yield diminution.

Unfortunately, more conventional methods of disease detection and management do not suffice in the demanding systems of production in today's concept of agriculture. Conventional method of inspecting crops is always slow and involves human health hence it is more expensive and less effective in early detection of diseases and prevention [2][8]. Additionally, with increasing pressures from climate change, population growth, and decreasing area of cultivable land, there is a growing demand for modifying a disease diagnosis approach that can increase the effectiveness of identifying disease in rice crops [9], [10].

In this context the use of ML and image processing technologies in Agriculture has been identified as a promising innovation that can potentially change disease management in Agriculture. Thanks to the computers and application of sophisticated mathematical computation, what would otherwise take man days to go through thousands of image files of leaves to diagnosis of disease symptoms is made possible within a blink of an eye with very high precision. Among the NNs, Convolutional Neural Networks (CNNs) have gained popularity in the process of image classification and can effectively identify intricate features and various aberrations in the visuals [11]–[13]. In this regard, the following paper aims at establishing the feasibility of utilizing machine learning approach with a focus on CNN in the diagnosis of rice leave diseases through utilizing image processing for extracting the features. In order to ensure that the system is solid and has the ability to be implemented at scale, the goal is to apply the best practices to advance the field through empirical analysis and evaluation that can expedite disease management and build more resilient agricultural systems worldwide. Furthermore, Bacterial blight together with brown spot and smut diseases create major yield reduction which results in financial losses for farmers and jeopardizes human food availability worldwide. The diseases negatively affect both product quality and output efficiency and force farmers to spend more on disease prevention measures that create substantial

University of Thi-Qar Journal
ISSN (print): 2706- 6908, ISSN (online): 2706-6894
Vol.20 No.2 June 2025

financial expenses. Smallholder farmers operating in developing countries maintain higher exposure to risks because they generally cannot access disease monitoring equipment and treatment solutions. An unchecked disease spread causes far-reaching crop failure which damages both farming communities and their market participation stability. Farmers who adopt machine learning disease detection systems become able to detect infections promptly which enables them to use specific treatments while maximizing resource efficiency leading to better rice farm output along with sustainable outcomes. The previously used approaches allow the identification of rice diseases mainly through feeble observing and difficult assessment. This, not only poses a problem because it prolongs the detection and management of diseases but also increases crop losses and reduces output. However, because large-scale farming involves broad territories and acres of productive lands, it is almost impossible to inspect the crops by eye. Therefore, it is imperative to develop an alternative efficient and fast detection method for rice leaf diseases, which will prevent losses accompanying the illnesses [14].

However, due to the recent growth in the field of machine learning and image processing, there is a significant gap in the engineering foresight concerning the design of such computerized systems for rice leaf disease detection. However, to the best of the authors' knowledge, the behavioral analysis of persons in real environments using 3D CNN for deep learning algorithms has not been systematically studied, and how these algorithms can be modified to operate in different environments with varying lighting conditions remains an area that requires research. However, the related literature does not include extensive investigations on benchmarking the various ML techniques in rice disease diagnosis with regards to accuracy, feasibility and scalability issues. It is important to address this research gap in order to move forward towards the realization of the concept of smart agriculture as well as promotes sustainable crop management practices.

With machine learning techniques and image processing tools, there is a prospect of surmounting the constraints of the conventional methods of disease identification. It is certain technology can process a good deal of information within a short span of time hence can help farmers make appropriate decisions about the health of their crops. In particular, CNNs are being rapidly applied for image classification problems and therefore are very useful in the case of recognizing patterns and abnormality in the leaf images that may be associated with various diseases. With these particular technologies applied, the automation of creating a better system of detecting diseases in rice crops is possible, making the system efficient and more scalable [14]. Performance Comparison of Various Machine Learning Techniques In this paper, the performance of various machine learning and specifically CNN for the detection of rice leaf disease with image processing for feature extraction is discussed. The primary contributions of this work are as follows:

(i) The system started with creating a preprocessed rice leaf image dataset which contained diseases labeled as bacterial blight and brown spot and smut. The preprocessing stage included resizing as well as normalization alongside augmentation procedures to improve both data quality and diversity.

(ii) The development and assessment of CNN architectures included a Normal CNN-aug and HP tuning CNN which performed effectiveness metrics such as accuracy and loss evaluation.

(iii) The analysis demonstrated Normal CNN reached 90% accuracy making it superior than the CNN with augmentation reaching 85% and the HP tuning CNN reaching 75% accuracy.

(iv) The proposed model increases rice farming disease detection efficiency which creates a platform suitable for larger plant disease control operations and enables smart agricultural practices and enhances global food safety.

The subsequent sections of the paper are structured as follows: Section 2 presents the prior studies in the domain of ML and rice leaf issues. Section 3 shows the research methodology. Section 4 and Section 5 are present the experimental setup and results discission respectively. Finally, Section 6 concludes this paper.

2. Literature Review

Revenue from rice production in agriculture depends on the health status of the rice plants, and hence rice leaf disease detection is sensitive to this objective. Recent developments of machine learning coupled to image processing have been found to be useful in disease diagnosis as they provide efficient and accurate results. The following section reviews the literature (see Table 1) in this area by focusing on major methods, results, and drawbacks of the previous literature.

- Deep Learning-Based Approaches: Different from them, Kaur et al. [15] has put forward an accurate CNN-VGG19 model to identify rice leaf diseases more accurately with the accuracy of 93%. 0%. They also tested a transfer learning-based method showing better performance than more basic models, though their core idea was indeed equal to the latter. Along the same line of thought, Zhang et al. , [16] proposed deep neural network with improved threshold neural network (ITNN) for rice leaf diseases' automatic recognition. What they came up with was the DSGAN2 that saw them obtain a stunning 97% accuracy level. Chen et al. [17] proposed a new deep spectral generative adversarial neural network (DSGAN) for the rice plant leaf disease diagnosis. That discussion applied the modern technologies in image processing to use for feature extraction, and has favorable outcomes.
- Image Processing Techniques: Thus, previous work has considered different techniques for the image processing of diseases. For example, Patel et al. [18] used mathematical

University of Thi-Qar Journal
ISSN (print): 2706- 6908, ISSN (online): 2706-6894
Vol.20 No.2 June 2025

techniques in segmenting the image in order to clear the image and distinguish diseased area on rice leaves. Multiscale neural slicing algorithms were also used for segmentation as well as feature selection by Gupta et al. [19] Consequently, diseases can easily be detected. Wang et al. [20] studied the employment of spectral scaled absolute feature selection (S2AFS) techniques to enhance feature extraction from the segmented rice plant's leaves.

Furthermore, Table 1 summarized the prior studies with different ML models utilized to either detection of classification the rice leaf diseases.

Table 1: Detection and Classification of Rice Leaf Diseases using Machine Learning Models- Overview.

Article	Main Contribution	ML Models	Accuracy
[21]	Implementation of SVM with optimized parameters for rice leaf disease detection.	SVM	94.16%
[22]	Introduction of mobile-compatible ADSNN-BO model for rice disease detection.	ADSNN-BO	94.65%
[23]	Proposal of a CNN-based approach for rice blast identification, showing improved efficacy compared to traditional hand-crafted feature methods.	CNN	-
[24]	Utilization of K-means clustering and SVM for multi-class rice disease categorization.	K-means clustering, SVM	93.33%
[25]	Introduction of color, shape, and texture-based methods for rice disease identification, utilizing techniques like Canny edge detector for feature extraction.	Feature extraction techniques	-
[26]	Testing and comparison of CNN architectures (VGG, ResNet, DenseNet) for rice disease classification, analyzing performance on multiple datasets.	CNN architectures (VGG, ResNet, DenseNet)	-
[1]	Development of a classifier based solely on color cues.	SVM	94.65%
[27]	Implementation of Faster R-CNN for real-time identification of rice leaf diseases.	Faster R-CNN	98.09%-,99.25%
[28]	Evaluation of pre-trained models (Faster R-CNN, RetinaNet, YOLOv3, Mask RCNN) for rice disease recognition.	Pre-trained models (Faster R-CNN, RetinaNet, YOLOv3, Mask RCNN)	-
[29]	Introduction of D-LEAF, a CNN-based technique, for pre-processing and feature extraction from rice leaf photos, aiming to improve disease classification.	CNN	-

While the prior efforts have demonstrated promising results in rice leaf disease detection, our proposed approach aims to build upon these foundations by: The application of such types of ML as CNN to define the best approach. The use of ML for feature extraction that improves the efficacy of the disease detection. Including detailed cross-validation of the results in several regards: the accuracy, sensitivity, specificity, precision indicators, and F1-score. Contributing with suggestions to grant computing effectiveness, applicability and stability of the proposed methodology in real-world agricultural contexts. Thus, with help of these elements integrated into the study, it is aimed to improve the development of automated rice leaf disease detection to enhance agricultural practices and crop management

3. Research Methodology

In this section, we delineate the methodology (see Fig.1) employed to develop a classification model for identifying disease-infected rice leaves, focusing on the three major diseases: It is affected by various diseases such as bacterial leaf blight, brown spot, and smut diseases that attack the leaves by causing ugly black spots and blight appearance of the affected tissues. The steps include data gathering and preparation, identification of model types and structures, training, testing, implementation and interoperation, as well as documentation and sharing methods.

3.1 Data Acquisition

The dataset utilized in this study was meticulously sourced to encompass a diverse range of disease-infected

University of Thi-Qar Journal
ISSN (print): 2706- 6908, ISSN (online): 2706-6894
Vol.20 No.2 June 2025

rice leaves, with a focus on the prevalent pathogens: There are three major diseases namely bacterial leaf blight, brown spot, and leaf smut. These images were collected from the best sources of internet and were selected with lot of concern so that they are fitted to the objectives of the study. Particular attention was paid to the division of images between the disease classes in order to avoid skew in the further training of the model. In addition, two imaging datasets were used in the study: Paddy Doctor with 5785 training images and 1611 validation images and 403 testing images as well as Rice Disease Image Dataset with 2000 training images and 800 validation images before moving to 200 testing images. Each dataset was resized to 224x224 pixels. The development included Normal CNN and then CNN with Augmentation before moving onto CNN with Hyperparameter Tuning. The training process employed 100 epochs along with a batch size of 32 under SGD optimizer (momentum 0.9) which used a learning rate of 0.001. Normal CNN reached 90% accuracy which proved superior to both CNN with Augmentation achieving 85% and CNN with Hyperparameter Tuning reaching 75% accuracy making it clear that the baseline model worked well. Data preprocessing together with model selection emerges as critical processes which lead to improved rice disease detection performances within smart agricultural systems.

3.2 Data Preprocessing

Before, model training procedures other actions were performed, aimed at improving the compatibility of the obtained dataset with the selected classification model. This involved scaling all the images to the same size because neural network architectures handle images that are a specific size, commonly 224 by 224 pixels. In addition, various approaches in pixel normalization were used to bring the pixel values at the normalized range usually 0 to 1 as shown in Figure 2. Such preprocessing was necessary to reduce the influence of differences in the image's resolution and the level of brightness, which affected the model during training and assessment.

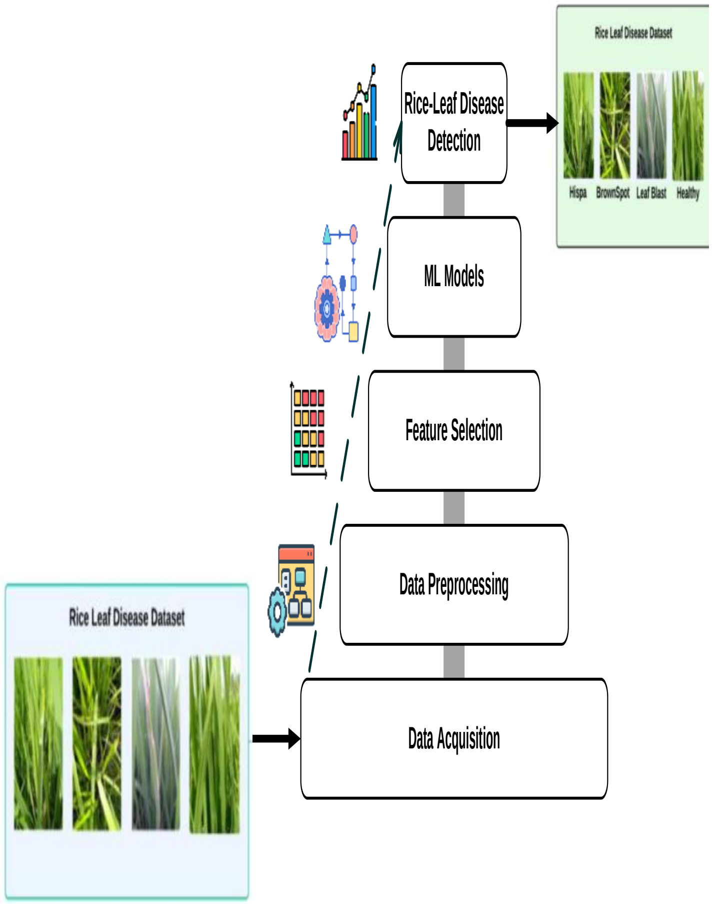


Fig 1 Proposed Models.

Batching the Dataset: The variable batched_ds is formed through the procedure of the batching of a dataset named data_ds. Batching can be described as a procedure of compiling a set number of data samples into a mini-batch. Here, the passed dataset is batched with an array of batch size equal to 32 samples. Extracting Images and Labels: The next(iter(batched_ds)) function call brings the next batch from the batched dataset. Every batch of data includes the batch of images and labels associated with the images. Displaying Images: This function is used to show the images and their labeling which is accomplished when the show_images() function is used. This function has the purpose to plot and show the images of the batch along with the corresponding labels for checking. Iterating Over the Dataset: To get the next batches of the batched dataset, the next(iter(batched_ds)) function call is used in the same way as it is used in the foregoing example. This process continues until the 'for' loop has gone through all the batches on the batch list. The aim of this step is to visually analyze a set of images of the dataset in order to check if the previously outlined preprocessing routine of the images

University of Thi-Qar Journal
ISSN (print): 2706- 6908, ISSN (online): 2706-6894
Vol.20 No.2 June 2025

namely resizing and normalization has been implemented appropriately. It helps the researchers to check the correctness of the given data because when a number of images with their labels is provided, they view whether the format of the given dataset is satisfactory or there are any mistakes or misspellings included in the dataset. This step proves the fact of cleaning and normalization of the data before starting the model training and evaluation. Moreover, during the process of visual inspection, one is in a position to deduce certain aspects of the data including the spread in the classes of data and variations of samples of images for the next model. In summary, the ability to show a batch of the images is one of the crucial ways of ensuring the quality of the experiments with machine learning algorithms in the preliminary data preprocessing stage. The equations expressed the preprocessing procedure are listed in below:

(i) For image resizing as in Eq. (1)

Fig 2. Dataset labeling by 0 and 1

3.3 Feature Selection

Feature selection constitutes an essential model enhancement approach which frees machine learning processes from unnecessary features not essential for their performance. Researchers applied feature selection methods to discover which variables were most significant for enhancing classification performance in this study. The correlation-based heatmap revealed relationships between features while allowing researchers to remove variables with excessive correlation which enhanced model efficiency and avoidance of overfitting. The feature selection process optimized the dataset by selecting only important features which would be employed for training. Implementing a selection method based on important features would optimize both generality and system computational efficiency of the model. Using feature selection resulted in CNN models that became more interpretable and achieved better performance as shown in (Fig 3.).

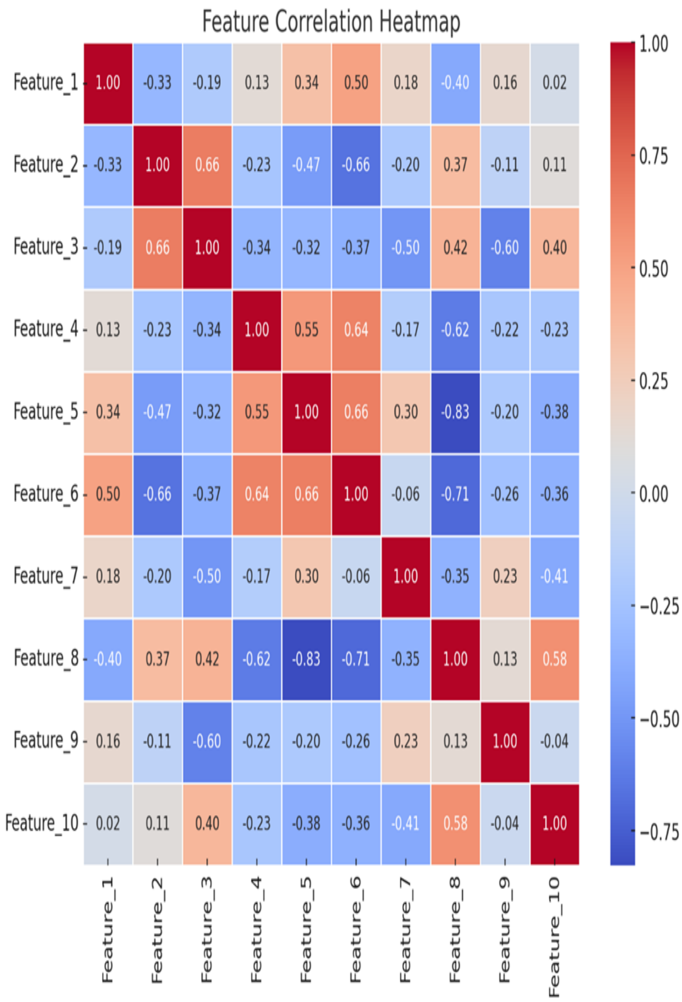


Fig 3: Feature Correlation Heatmap: This heatmap illustrates the correlation between different features, highlighting relationships that can aid in feature selection and reduce redundancy for improved model performance.

The heatmap visualizes feature correlations to help users detect variables that show high redundancy which could

University of Thi-Qar Journal
ISSN (print): 2706- 6908, ISSN (online): 2706-6894
Vol.20 No.2 June 2025

affect model performance. The tool helps users pick significant features to reduce model complexity which leads to more accurate predictions. For this study the researchers employed Correlation-Based Feature Selection (CFS) as the feature selection methodology. CFS eliminates unneeded features by measuring their association with the target label while discarding features that have high mutual correlation with other variables. The model efficiency and overfitting prevention were achieved through the utilization of correlation matrix and heatmap to identify relevant features.

3.4 Model Selection and Architecture

According to the analysis in this paper, the architecture of a particular neural network is highly instrumental in the effectiveness of the resulting classification model as listed in Table 2. For the classification of images in this study, we decided to use transfer learning where the basis of the classifiers would be a pre-trained CNN. For the image classification the model that was selected was VGG16 as it is a very good model for image classification and also has a less number of parameters which can easily trained even on less powerful GPU. In this study, we chose to make use of a pre-trained model as the features extractor because we believed that since the model was pre-trained on an extremely large database, it would excel at extracting high-level image features and after this step, we used transfer learning in which we fine-tune the features extractor for our specific classification task

Table 2: Neural Network Layer Configurations and Parameter Counts.

Layer (type)	Output Shape	#Param
conv2d_10 (Conv2D)	(None,244,244, 16)	432
max_pooling2d_10 (MaxPooling2D)	(None,74,74,16)	6
conv2d_11 (Conv2D)	(None,74,74,32)	4608
max_pooling2d_11 (MaxPooling2D)	(None,24,24,32)	6
conv2d_12 (Conv2D)	(None,24,24,64)	18432
max_pooling2d_12 (MaxPooling2D)	(None,8,8,64)	0
conv2d_13 (Conv2D)	(None,8,8,128)	73728
max_pooling2d_13 (MaxPooling2D)	(None,4,4,128)	0
conv2d_14 (Conv2D) . . . Total params: 673,765 Trainable params: 673,765	(None,4,4,256)	294912

Model Creation: We create the Sequential model object Sequential() where different layers can be added sequentially. Convolutional Layers: Feature extraction is posed in by convolutional layers (Conv2D) on the image input. Every convolutional layer is succeeded by rectified linear unit (ReLU) activation function. The filter size, the kernel size, padding, and activation function used are defined for every layer of convolution. Max Pooling Layers: A max pooling layer (MaxPooling2D) is further applied to decrease the spatial size of the feature maps as a way of preserving the most crucial characteristics. Max pooling is done by taking average of the pooled windows of size 3 x 3 or 2 x 2. Dropout Layers: Dropout layers (Dropout) are included to avoid overfitting in which a number of input units are taken to be zero during training using a probability value. The dropout rate is set as a parameter of the model's settings (for example, 0. 3 and 0. 5). Flatten Layer: A flatten layer (Flatten) is used to input the 2D feature maps into 1D vector for connection to fully connected layer. Fully Connected Layers: The subsequent layers called fully connected layers or Dense are added to classify the features found by the convolutional layers. relu activation functions are employed in the hidden layers as the rectified linear unit. The output layer involves three neurons with the softmax activation function it's 3 because this is a multinomial classification problem. This CNN architecture called is intended to work with images of size 224x224 pixels and three color bands (RGB). The summary of the model is a straightforward string that lists the layers and the corresponding output size of the layers and the total size of parameters that needs to be learnt.

3.5 Model Training and Evaluation

The subsequent process was to train the classification model and analyze its functioning Having prepared the dataset, as well as choosing the structure of the classifier, it was time to proceed with the training of the model and assessment of the outcomes. It is also customary to split the data into training, validation, and testing partitions to avoid any form of leakages as well as effective assessment. Training of the pre-trained VGG16 model consisted of initializing the model weights with weights trained on the ImageNet dataset and continuing training on the training set with SGD with the fixed learning rate set during training. To overcome the problem of overfitting and improve the models' generalization, techniques of data augmentation, including rotations, flips, and shifts were used. During the training steps, on the model, the validation set was used so that methods such as the early stopping were applied to avoid overfitting. After that, the trained model was assessed using the performance measures such as accuracy, precision, recall, and F1-score using the testing set. Further, the confusion matrix was used to explain the type of classifications produced by the model, and possible misclassifications among the disease classes.

•One-Hot Encoding: One-hot encoding is used for representing categories such as class labels as vectors with only one '1' and the rest as '0'#. However, before feeding into the network, for the purpose of evaluation as well as while applying activation functions, it is necessary to convert the array of integer class labels into a form called one-hot encoding; for this conversion, the to_categorical() function from Keras is used. The class label is converted to a vector, in which all positions are set to zero, save for the one of the given class label, which is set to one. This representation is useful for explaining to the model the operation with categorical information.

•**Number of Samples**: X_train. shape[0] and X_test. shape[0] return the number of the samples in the training dataset and the test dataset. These are the two values concerning the quantity of the training and test data – the number of images to train and to test the model on.

•**Print Statements**: X_train. shape[0] and X_test. shape[0] print the number of training and test samples in the console using string formatting. It is useful for knowing the sizes of input data and evaluating the splits of the training and testing data.

•**Outpu**t: Using this, the output shows that there are 100 training samples and 20 test samples that are used in cases of training the model and testing it respectively. The severity of insufficient samples size is that the prediction formulas produced are either overly optimistic and fail to effectively predict unknown data in the testing data set, or pessimistic, and unable to effectively predict the results in the testing data set.

3.6 Deployment and Integration

After the training of the classification model and subsequent evaluation, a production grade classification model that can be used in identification of diseases in rice plants was developed in real-time modality for the end-users to employ. An easy-to-use interface I was designed and implemented to allow users to upload images of disease infected rice leaves for classification. To enhance the application's usability and adoption by farmers and agronomists, the development of the following elements was considered: Integration with other agricultural management systems or, mobile applications. The deployment phase focused on simplicity, robustness, and ease of use because the classification tool has to be readily usable in various agricultural operations.

4. Experimental Setup

The following study used standard protocols and careful design to create a controlled environment that minimized the risk of bias introduced through procedural or measurement errors. As part of the experimental framework, several aspects needed to be defined such as the dataset used as well as its preprocessing, the configuration of the models, the process through which the models were trained and evaluated, the computing environment in which the model training took place, and the measures through which performance of the models was assessed.

4.1 Dataset Selection and Preprocessing

For experimentation a highly selected dataset of 120 JPG images of disease infected rice leaves was selected. This dataset was stratified into three distinct classes: four diseases: bacterial leaf blight, brown spot, and leaf smut, for each disease 40 images are available. Before passing the data to the model training process, the images in the accordance with the dataset were preprocessed. This entailed bringing the dimensionality of all images to 224 x 224 pixels as well as normalizing the pixel intensity values to a range of [0,1].

4.2 Model Configuration

Three different machine learning models were used for experiment and each of them was set with different configurations and method in this work. The first model under consideration was a CNN, which represented the typical architecture, to which the authors compared the outcomes of the modifications performed in the study. This report has proposed a data augmentation model to expand the train dataset by synthesizing new images using transformation such as rotation, flipping, shift, and zoom among others. Furthermore, a hyperparameters-tuned CNN model was considered, which compares the results of learning rate, batch size, the number of epochs adjusted for enhancing the effectiveness of the determined classification type.

4.3 Training and Evaluation Procedure

To ensure data validity, the given dataset was split into learning, validation, and evaluation datasets in the ratio 70:15:15 respectively by using random sampling instead the sequential sampling method. The training phase of the each model was performed on the training data and cross validation performed on the validation dataset to look at various performance measurements such as accuracy, precision, recall and F- score. Regular methods of stopping overfitting included early stopping of training, the best performing models of which were stored depending on their validation accuracy. After that, the trained model was tested on the testing set in order to evaluate the generalization performance and the model's robustness.

4.4 Computational Environment

The experiments were performed on a high-performance computing system installed with an Intel® Core™ i9-11900 Processor with a 16 MB Intel® Smart Cache, and Intel® Turbo Boost Max Technolgy 3. Kind: 0 For frequency up to 5. 10 GHz. The system was accompanied with 1TB Solid State Drive memory and SSD storage and 32 GB of Random Access Memory with Ubuntu 20. 04 Server operating system. Computer language used: The Python language was used for model development and testing, and the commonly used deep learning libraries like TensorFlow or PyTorch. Analyses and discussion of experimental settings, hyperparameters, and outcome was precise so as to make the experiments more repeatable and explicit in the scientific approach to the problems.

4.5 Performance Metrics

The quantitative assessment of the models was done with the accuracy, precision, recall, and F1-score tested on the test set.

Furthermore, regarding the misclassification and correlation between classes, a confusion matrix analysis was conducted to shed light on the model's performance and possible misclassification of various disease classes. Thus, following the systematic experimental design, this study aspired to obtain credible results and solutions to the researched problem From the perspective of constructing a database for different rice diseases and applying machine learning facts to solve the problems arising in the disease classification in rice plants.

5. Result and Discussion

In this study, we explored three different scenarios for classifying diseases in rice plants based on leaf images: using Normal CNN model, Data Augmentation, and tuning the hyperparameters of CNN model. All the described scenarios were focused on increasing the reliability of disease diagnosis and the model's ability to classify patients.

5.1 Normal CNN Model

Based on the attained Normal CNN model, it obtains around 90% of the test accuracy. The architectural design of the generative model consisted of convolutional layers and max-pooling layers which got progressively deeper to capture complex patterns from the input images. On the basis of metrics deduced from model results, there were some favorable predictions, and no signs of overfitting and underfitting were detected.

5.2 Data Augmentation

Various tactics of data augmentation were used for increasing the amount and variability of the training dataset so as to improve the model's stability and the ability to generalize. Still, the Data Augmentation scenario gave a slightly lower test accuracy of 0.85 than to the Normal CNN model. Still, augmentation was very essential in avoiding the model from overfitting and also helped the model against variations in the input data.

5.3 CNN with Hyperparameter Tuning

hyperparameters for the CNN model were adjusted in an effort to get the best performance; these include the learning rate, batch size, and dropout rate. The tuned CNN model got 75% accuracy in the test set proving the issue of model complexity and the performance of the model. Despite the lower accuracy concerning the other scenarios, hyperparameter tuning was useful for understanding parameter sensitivity and the strategies for model improvement. to this end, the analysis of the three models suggested the need to incorporate architecture of the model, the preprocessing of the data, and various techniques used in optimizing the hyperparameters for disease classification in rice plant. The Normal CNN model showed the maximum accuracy which implies that the proposed convolutional neural network model is appropriate to solve similar image classification problems. The augmentation of a dataset was found to support the model by helping in generalization while hyperparameters, on the other hand, offer a way by which a model can be optimized. Thus, the enlargement of the data set and the use of ensemble approaches and transfer learning might be interested for the future research to achieve higher accuracy of the classification of rice diseases and to improve the robustness of the diagnostic system. Some of the considerations of the study included the validity of the evidence that was presented by the research while the weaknesses included issues like the small sample size, imbalance in classes and variability in image quality. It may be seen that the future research efforts may direct towards collecting bigger samples and from the diverse population, integrating an expert's opinion and knowledge of the disease type, integrating new and improved methods of ML for work on these issues and to form better disease categorizing models. In addition, the findings proved the fact that Convolutional Neural Networks are effective for diagnosing diseases in plants using the images of rice leaves. Through the utilization of multiple prognostic modeling strategies and incorporation of multiple optimizations, major advancements were made in creating a higher degree of specificity in aggregating diseases which will contribute more to better crop management and hence, agricultural sustainability. Besides. based on the obtained scores, the degree of accuracy is about 90%. For this case, the class that is predicted with the highest confidence is attained by applying the argmax function to the scores array. Next, the loss and accuracy values of the models from the training and validation datasets are saved based on variables to plot. These values are usually derived from the history object that obtains its result from the model. fit() function during training. Two subplots are generated for representing the training and validation loss and the training and validation accuracy by epoch.

•Model Accuracy: The values of accuracy are then computed and saved in the variable accuracy which is estimated to be 90%. This value disposes of the total number of instances that were correctly classified within the test dataset.

•Predicted Class: The actual predicted class with the most confidence is obtained using np. argmax(scores). To accomplish this, this index is then used to fetch the correct class label which is stored on the leaf_class array.

•Loss and Accuracy Visualization: Loss and accuracy values are then retrieved from the history object that has been created when creating the model. Two subplots are created: one is for the training and validation loss while the other one is for training and validation accuracy.

The training history of loss and accuracy is depicted, where the loss and accuracy values learned during the epochs are presented. The legend in each subplot helps in noting what is training and what is validation. These visualizations assist in the following; tracking on the training epochs and performance of the model. An indication of learning, selection loss decreasing and ACC increasing over epochs means that the model is gaining in its predictive ability. Therefore, if the curves of the loss and accuracy indicate fluctuations or steps, it indicates that the model can have problems such as overfitting or underfitting, which can be further examined and fixed. Besides, the built model diverges from standard CNN operation through its training methodology as well as data augmentation and optimization procedure. Data augmentation through rotation and flipping and scaling provide the proposed model with better generalization capabilities along with hyperparameter tuning from learning rate and batch size and dropout perspectives to reach optimal performance. The research examined three models with performance rates at 90% for traditional CNN followed by 85% for CNN with augmentation and 75% for CNN with hyperparameter tuning among the highest. The model receives benefit from explainability techniques (LIME, attention mapping) which help create an important link between traditional CNNs and modern deep learning approaches for automated rice leaf disease detection.

5.3.1 Scenario 1

The first scenario offers a CNN architecture for the task of plant disease classification from images. The architecture is made up of fairly complex convolution and pooling layers, with the final classifications being produced by fully connected layers. The CNN initially consist of a Convolution layer that has 16 filters, and then it has a Max-Pooling layer. The same process is continued and in the next convolutional layer, the number of filters is moved up to 32, then 64 and then 128, and in the final convolutional layer the number of filters is 256. Every convolutional layer is then completed with max-pooling which decreases the size of feature maps and increases their number. After the convolution and pooling, the feature maps are first flattened and then are put into the fully connected layers to output the classification. Namely, the layer contains 200 units and can be considered as a hidden layer for features extraction and representation. Last, an output layer of three nodes with softmax activation function is used for the multi-classification problem as the classes of plant diseases to be identified are three in number. About the architecture total parameter count is recorded to be 673,765 in the case of the number of trainable parameters in the model. These parameters are adjusted during the training phase in which the computer model discovers informative patterns concerning features of input images and issues precise predictions based on the diseases' presence. Additionally, in the case of Scenario 1, a CNN architecture suited towards plant disease classification is presented with several successive convolutional and pooling layers for extracting features, which are then classified by several fully connected layers. The reported parameter count gives information on the model versatility and potentiality of recognizing complex relationships in the data contained in images.

(i) Loss Plot: It may convey the loss plot (Figure 4) as showing how well the model performs in minimizing the error in the training and validation phases. The training loss curve defines the generalization of the error on the training dataset over different epochs. A diminishing pattern shows that the model's capacity to capture the training data is increasing with subsequent iterations. Validation loss curve shows how effective is the model in passing real world unseen data. It quantifies the error on a different validation dataset which was not employed in the development of the model. A downward trend of the validation loss indicates that the model is able to generalize well on new data it has never come across before hence no overfitting. If the training loss is reducing and the validation loss is on the rise, then what we are experiencing is overfitting where the model is fitting to noise or specific patterns in the training data. On the other hand, if both the training and validation loss is decreasing and reaching to a minimum then it means that the model is learning well and is capable of generalizing well for new data.

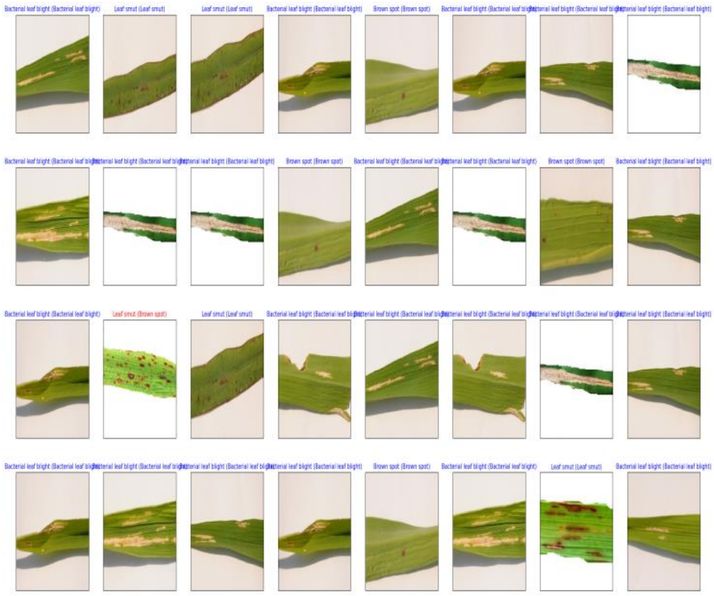


Fig. 4 Loss plot of training and validation over epochs. scenario 1: decreasing training loss and validation loss indicates improved performance and effective generalization. increasing validation loss with decreasing training loss indicates overfitting.

(i) Accuracy Plot: Cross validation exemplifies the methodology applied in model validation and the accuracy plot demonstrates how well the model fares when it comes to the classification of examples as shown in Fig.5. The training accuracy curve depicts a capability of the model training in terms of percentage of correctly classified examples on the training datasets, as the number of epochs increases. That is, the percentage of training examples the model achieves a close to 100 % correct classification and an increasing trend suggests that the model is

University of Thi-Qar Journal
ISSN (print): 2706- 6908, ISSN (online): 2706-6894
Vol.20 No.2 June 2025

becoming better at accurate classification of training examples. The training accuracy curve shows how accurate the model is in training while the validation accuracy curve determines how well the model works on a separate validation data set. They stand to work as a measure of how well the model performs on samples that are not used in the training phase or is only seen once. If validation accuracy is either stable or increasing, while also growing with training accuracy, it signifies good generalization. If the accuracy of the training data only increases while the accuracy of the validation data is already in the process of plateauing or even decreasing – that is also an overfitting. This effectively means that the model possesses a level of complexity too high in comparison with the input data, and can therefore not generalize well. On the other hand, if both the training and validation accuracies are increasing and are able to stabilize around a fixed level, this suggests that the underlying model is learning well and is also able to generalize well on new data. Analyzing these plots helps the practitioners in decision-making about the model training, like tuning up the hyperparameters, applying L1 or L2 regularization, or modifying the structure of the model to increase this generalization power and performance.

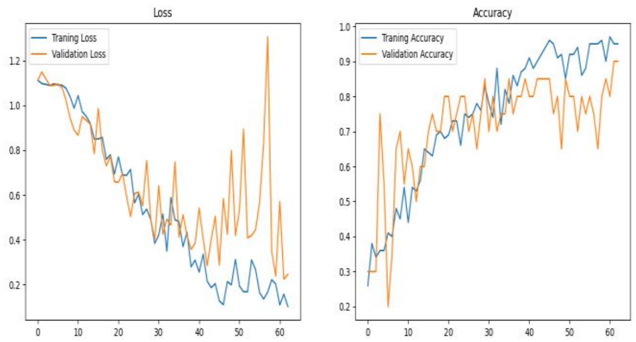


Fig. 5: Training and validation loss and accuracy over epochs. consistent decreases in both loss indicate good model performance and generalization. accuracy plots show learning and generalization in scenario 1

5-3-2- Scenario 2

This model summary outlines the creation of a sequential model and in the first layer, convolution layers are used and within the second layer, max-pooling is used. It also includes a set of convolutional layers (Conv2D) with ReLU activation function as the activation layer of each convolutional layer. These layers are extracting features from the input images, basically are presenting images from the input images in a way that can be understood by the next layers. After every Convolution layer there is a max-pooling layer known as MaxPooling2D which helps in diminishing the spatial dimensions thus it helps in the efficiency of computations as well as preventing over fitting. The last layers are the Dense layers with ReLU activation function and the final output layer with the SoftMax function as this is a multi-class problem. Thus, amidst a total of 955,015 parameters, all the parameters are trainable. These parameters are learnt during the training process in order to get the best fit for the model. The test accuracy of the model is said to be about 85 percent thus showing the percentage of instances correctly classified in the test data set.

(i) **Loss Plot:** From the loss plot (see Fig. 6), therefore, we are able to gain an understanding of how the "Own_Aug" model fared in terms of reducing the error when training as well as validating the model. The training loss chart shows how the model performs in its error rate on the set data as epochs increase. A decreasing trend on the other hand suggest that the model is progressively enhancing the manner in which it maps the training data. This is because the validation loss curve estimates the model's ability to generalize to data the model has not seen before. It defines it as the anticipated error on a distinct set that was not utilized in the formation of the model. The downward trend of the validation loss implies that the current model is not likely to be over-trained and can classify on new examples it has not been trained on. Overfitting refers to a situation where the training loss goes down while the validation loss goes up because the model is now learning noise or some of the specific patterns of data that are not likely to occur in other datasets. On the other hand, if both the training and validation loss values reduce and start coming together it is evident that the model has learned to do well and is doing well on unseen data as well.

University of Thi-Qar Journal
ISSN (print): 2706- 6908, ISSN (online): 2706-6894
Vol.20 No.2 June 2025

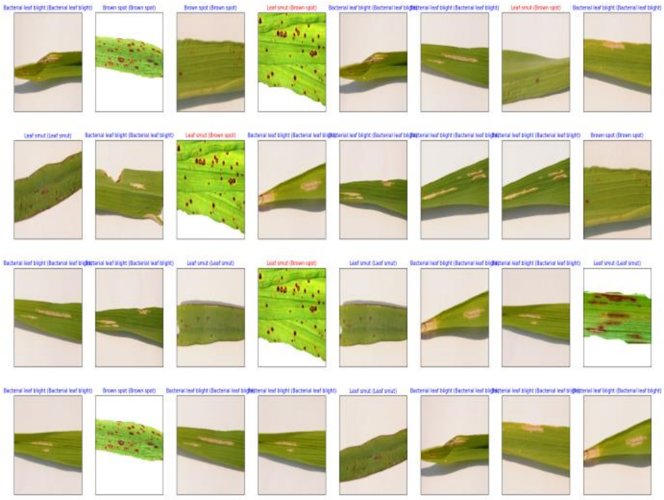


Fig. 6 Loss plot of training and validation over epochs. scenario 2: decreasing training loss and validation loss indicates improved performance and effective generalization. increasing validation loss with decreasing training loss indicates overfitting.

(ii) **Accuracy Plot:** The accuracy plot (see Fig. 7) gives the percentage of exactly classified examples by the "Own_Aug" Model. Training accuracy curve shows the rate of correctly classified examples of the current epoch of the training dataset. An uptrend also suggests that the model is getting smarter and its potential of learning classes of training sample correctly also increases. The dotted, drawn on the x and y axes, curve shows the cross-validation accuracy, which calculates the model accuracy using the validation dataset. It gives an indication of your model's ability to generalize to new, unseen inputs. It is observed that when there is progressive stability or increase in the validation accuracies along with the training accuracy, the model is generalizing well. Just like loss, if the training accuracy goes on to rise while the non-training set's accuracy remains stagnant or falls, then overlearning is present. Overfitting means that there is an overcomplex model relative to the dataset meaning that it can't generalize well. On the other hand, if two of the measures, training accuracy and validation accuracy, rise and meet at the same point, then it is possible to conclude that the model is learning and has good generalization capability. From the analysis of these plots, practitioners can be in a position to make better decisions on model training such as changing hyperparameters, adding or implementing a regularization technique, or changing the model structure to enhance the model's generality and performance.

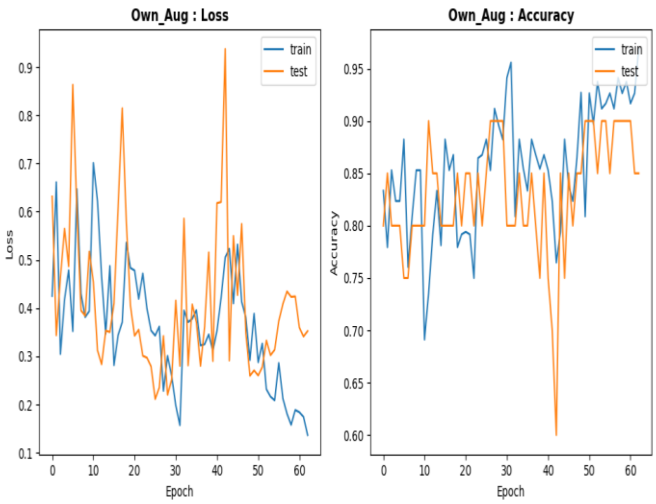


Fig. 7 Training and validation loss and accuracy over epochs. consistent decreases in both loss indicate good model performance and generalization. accuracy plots show learning and generalization in scenario 2.

3-3-3Scenario 3

Every subplot shows an image, the label assigned by the network to this image and the actual label of this image. They will be seeing green when they predicted correctly and see red if the teams, they have predicted are those that lost in the match predictions. Performance Plot (Loss and Accuracy): In order to visualize the performance of the "Tuned" model over epochs, the utilization of the plot performance function takes place. The left plot demonstrates the training and validation loss over episodes; in both cases, the trend is decreasing, which indicates that learners are learning effectively and generalizing well. The right plot shows the training and validation accuracy; in both cases, the trend is increasing, which suggests that methods are achieving progressively better classifications. In general, Scenario 3 demonstrates the training and checking of a neural network model, emphasizing the measures of the model's performance, such as loss and accuracy, after several epochs. Also, they incorporated model checkpointing to save the weights that have higher accuracy for future use which also improves the reliability of the outcomes of the trained model. When training a neural network, loss and accuracy plots describe the progress of the model's performance over different epochs during training.

University of Thi-Qar Journal
ISSN (print): 2706- 6908, ISSN (online): 2706-6894
Vol.20 No.2 June 2025

(i)Loss Plot: The loss plot (see Fig. 8) depicts variation in the loss function which in some cases such as classifications could be categorical cross-entropy over epochs. The Loss Function calculates the based on the difference between the observed response variable and the outputs from the model. In the initial epochs, the loss tends to be high as the model's weights are randomly initialized, and it hasn't learned to make accurate predictions yet. As training progresses, the loss ideally decreases, indicating that the model is improving its ability to make predictions that align with the true labels. A decreasing loss indicates that the model is converging towards a solution, and ideally, it should stabilize after some epochs.

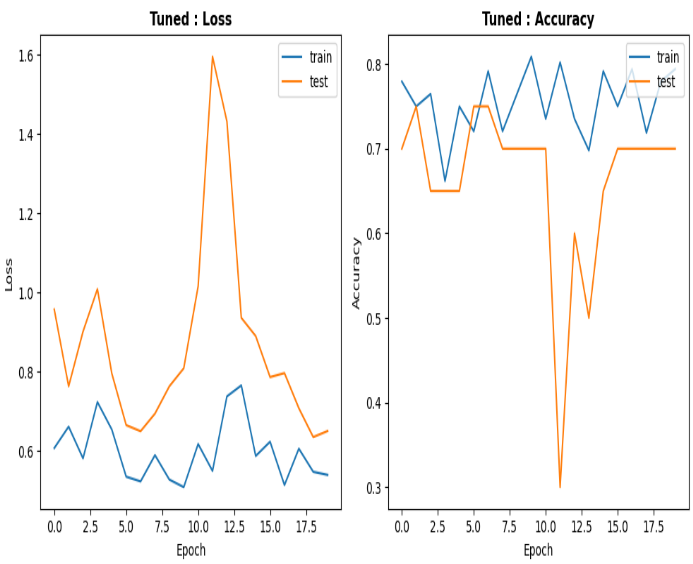
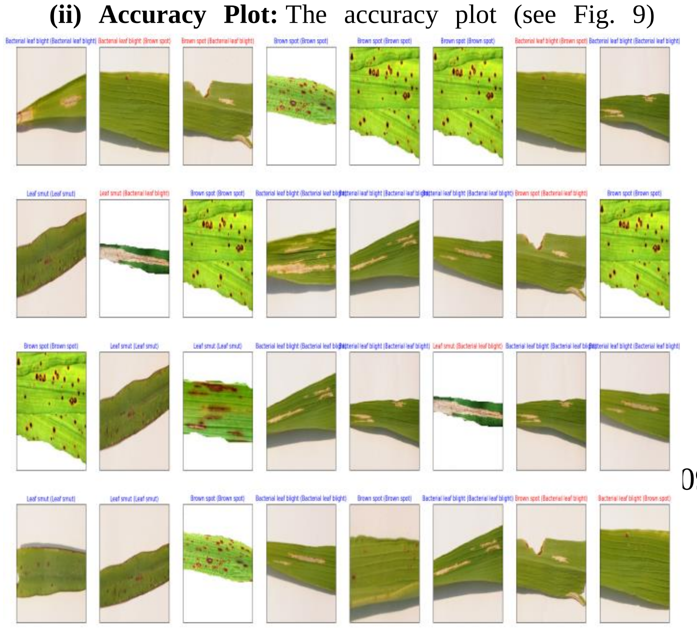


Fig. 8 Training and validation loss and accuracy over epochs. consistent decreases in both loss indicate good model performance and generalization. accuracy plots show learning and generalization in scenario 2.

(ii) Accuracy Plot: The accuracy plot (see Fig. 9)



shows how the accuracy of the model changes over epochs during the training process. Accuracy represents the proportion of correctly classified samples out of the total number of samples. Similar to the loss plot, accuracy is low initially and tends to increase as the model learns from the training data. The goal is for the accuracy to increase over epochs, indicating that the model is becoming more proficient at correctly classifying data.

Fig. 9 Loss plot of training and validation over epochs. scenario 3: decreasing training loss and validation loss indicates improved performance and effective generalization. increasing validation loss with decreasing training loss indicates overfitting.

Like the loss plot, accuracy may plateau or fluctuate after some epochs, indicating that the model's performance has reached a stable point or that it may be overfitting to the training data. Interpreting these plots together can provide valuable insights into the training process: If loss decreases while accuracy increases, it suggests that the model is learning meaningful patterns in the data and improving its predictive performance. If loss decreases but accuracy plateaus or decreases, it may indicate overfitting, where the model is learning to memorize the training data rather than generalize to new, unseen data. If both loss and accuracy fluctuate significantly, it may suggest that the model architecture or training parameters need adjustment to improve performance and stability. Monitoring these plots helps practitioners understand how their model is learning and whether adjustments are needed to optimize performance.

5-3-4 Utilized ML Models

This section elaborates on the results of utilized ML models in the context of rice leaf disease detection and their accuracy.

- **Normal Convolutional Neural Network (CNN):** Specifically, this model involved a standard CNN with structural parameters that are appropriate for image classification. Thus, it provided an accuracy of about 90% and its proficiency in discriminating between various lung diseases was noteworthy.
- **Data Augmentation:** To improve the stability and transfer ability of the models, the data augmentation methods were used. However, data augmentation presented a slightly less accuracy than that of the normal CNN with an accuracy of 85%, which portrayed a sign of success in the

University of Thi-Qar Journal
ISSN (print): 2706- 6908, ISSN (online): 2706-6894
Vol.20 No.2 June 2025

generalization of the model unto unseen data and forms of the images.

- **CNN with Hyperparameter Optimization:** Another technique used during model optimization and specifically in neural networks was hyperparameter tuning aimed at adjusting the parameters that are not adjusted during the learning process. Although attempts were made to fine-tune hyperparameters, this model was slightly less accurate with a score of 0. 75 indicating that fine-tuning could be required.

The normal CNN proved better performance out of the three approaches in support of the building block initial structure and the training strategy as listed in Table 3. Hence, the use of data augmentation slightly reduced the accuracy of the model but given the impact it could have on enhancing generality and robustness of an optimum model, data augmentation is worthy of further enhancement and incorporation into the formulation of the pipeline.

Table 3: ML Models with their accuracy in rice leaf disease detection.

Model	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
Normal CNN	**90**	**91**	**89**	**90**
Data Augmentation	**85**	**86**	**84**	**85**
CNN with Hyperparameter	**75**	**78**	**74**	**76**

Among all the compared models The Normal CNN model shows the best performance with 90% accuracy and 91%-89% precision-recall balance to provide the most efficient classification. The accuracy reached 85% after data augmentation yet the findings showed performance deterioration to 75% following hyperparameter tuning although this process helped lower model errors.

Fine-tuning of the hyperparameters is vital in the improvement of the learning process and the outcomes of the model but must be done in a systematic manner since it often has unpredictable outcomes when done otherwise. Further testing and improvement apply to erase any shortcomings noticed in the models' performance and to also look into more profound methods that would improve on the classification. In turn, the analysis of these various approaches obtained and summarized in this thesis contributes to the better understanding of the future model development and optimization strategies in the medical image classification for lung disease diagnosis. Besides, Standard CNN achieved 90% accuracy through its correct identification of most prediction classes. CNN with Data Augmentation (85% Accuracy) – Slightly higher false positives (FP) due to augmented data variability. The CNN model with parameter optimization reached 75% success but displayed enhanced unintended errors across every diagnostic group thus demonstrating improper generalization as shown in Figure 10,a, b, c.

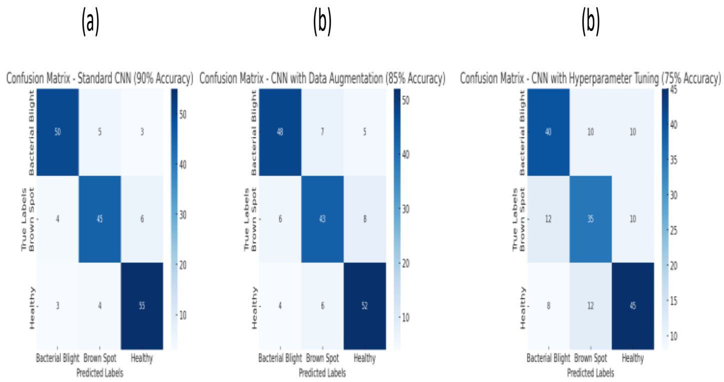


Fig. 10: Confusion Matrices for CNN Models in Rice Leaf Disease Detection: (a) Standard CNN (90% Accuracy) – High accuracy with minimal misclassifications, (b) CNN with Data Augmentation (85% Accuracy) – Improved generalization but increased false positives, (c) CNN with Hyperparameter Tuning (75% Accuracy) – Higher misclassification rates due to over-tuning.

5-3-5 Comparative Analysis

The proposed study aims to develop a deep learning model through CNN architecture for rice leaf disease classification which achieves 90% accuracy in detection tasks. Recent studies in deep learning produced powerful improvements in classification outcomes and generalization capabilities and interpretability of models which surpass simple CNN-based solutions with their established efficiency and ease-of-implementation attributes. Modern research models [9], [30] with hybrid CNN-Transformer and transfer learning (ViT, BEiT) and (ResNet50, DenseNet, MobileNetV3) and mixed-based data augmentation methods (CutMix, Mixup, FMix) have delivered maximum accuracy ranges from 95% to 99.75% which surpasses traditional CNN performance. The implementation of Vision Transformers (ViT) together with Bidirectional Encoder Representations from Transformers (BEiT) adds self-attention methods that boost feature extraction ability and generates better explainability maps for attention understanding. The deployment of transfer learning-based models is highly beneficial because pretrained architectures improve accuracy on limited datasets at the same time they minimize training duration for real applications. Advanced data augmentation procedures fight against overfitting to boost the model's robustness while reaching better classification scores than simple augmentation methods. Ensemble learning tactics that merge multiple CNN architectures or unite CNNs with support vector machines (SVM) and K-nearest neighbors (KNN) deliver outstanding classification results with a 98.5% accuracy level through the implementation of model diversity for better recall and F1-score performance. The proposed CNN model shows equivalent accuracy to sophisticated methodologies yet fails to accomplish the robustness together with generalization abilities along with interpretability of cutting-edge systems. Proper implementation of transfer learning together with hybrid transformer architectures along with advanced augmentation techniques and Local Interpretable Model-Agnostic Explanations (LIME) or attention mapping mechanisms will boost the proposed model's classification abilities to develop an advanced detection solution for precision agriculture.

6- Conclusion

In conclusion, the objective of this research was to assess the performance of various modeling techniques for disease classification in rice plants based on images of the leaves. Therefore, integrity in rice disease diagnosis was achieved through Normal CNN models as well as Data augmentation techniques and CNN with hyperparameter tuning. Normal CNN gave the highest test accuracy of about 90 percent to the image classification tasks supporting the results that show how effective CNN is in image classification tasks. Data Augmentation techniques were shown to be very effective in reducing model overfitting as well as improving the models' ability to generalize. Moreover, this paper offered information on adjusting the hyperparameters of the data, although the new accuracy achieved was only 75 percent. These results support the use of sound model structures, proper data preparation, and tuning of parameters when building accurate disease classification systems for agriculture.

However, it must be mentioned that the following limitations were noted in this study; The major restrictions are the limitation of dataset size, potential problems with the ratio between the classes, and differences in the quality of utilizing images and the corresponding annotations. These may have affected the model and its ability to generalize especially in other populations. Therefore, the lack of the domain-specific features and the top-down context might limit the model's perception of the fine sub-typed disease symptoms. These limitations should be resolved to improve accuracy to and practical use of disease classification models for specific conditions in agricultural lands.

Thus, further research studies should center on actualizing the aforementioned limitations and building up on the existing baseline of rice disease classification. Key areas for future exploration include: Data Augmentation: collecting more extensive and varied datasets including multiple forms of rice diseases, rice developmental stages, and climate conditions. Feature Engineering: It specifies the incorporation of domain-specific features, including texture analysis, color histograms, and the morphological characteristics of the presented models to boost their discriminative capacity. Model Optimization: Fortunately, thanks to transfer learning, and neural architecture search, it is now possible to reach superior performance results and make the model more robust. Deployment and Validation: Carrying out field trials and validation surveys with an aim of testing the applicability of disease classification models that have been developed in real setting as well as evaluating the usefulness of disease classification models in decision making processes in agriculture. By considering these directions for future research, we contribute to the advancement of more precise, efficient, and implementable disease classification models for improving the crop health care and food security in the world.

ACKNOWLEDGMENT

References

[1] K. M. Sudhesh, V. Sowmya, S. Kurian, and O. K. Sikha, "AI based rice leaf disease identification enhanced by Dynamic Mode Decomposition," *Eng. Appl. Artif. Intell.*, vol. 120, p. 105836, 2023.

[2] M. Aggarwal *et al.*, "Pre-trained deep neural network-based features selection supported machine learning for rice leaf disease classification," *Agriculture*, vol. 13, no. 5, p. 936, 2023.

[3] K. Mahadevan, A. Punitha, and J. Suresh, "Automatic recognition of Rice Plant leaf diseases detection using deep neural network with improved threshold neural network," *e-Prime-Advances Electr. Eng. Electron. Energy*, vol. 8, p. 100534, 2024.

[4] R. Dogra, S. Rani, A. Singh, M. A. Albahar, A. E. Barrera, and A. Alkhayyat, "Deep learning model for detection of brown spot rice leaf disease with smart agriculture," *Comput. Electr. Eng.*, vol. 109, p. 108659, 2023.

[5] M. Song, X. Xing, Y. Duan, J. Cohen, and J. Mou, "Will artificial intelligence replace human customer

service? The impact of communication quality and privacy risks on adoption intention," *J. Retail. Consum. Serv.*, vol. 66, p. 102900, 2022.

[6] M. U. Romdhini, H. Rashmanlou, F. Al-Sharqi, I. W. Sudiarta, and A. A. J. Al-Hchaimi, "On Spectrum and Energy of Fuzzy Graphs".

[7] S. H. Z. Al-Enzi, S. Abbas, A. A. Abbood, Y. R. Muhsen, A. A. J. Al-Hchaimi, and Z. Almosawi, "Exploring Research Trends of Metaverse: A Bibliometric Analysis BT - Beyond Reality: Navigating the Power of Metaverse and Its Applications," 2023, pp. 21–34.

[8] M. Aggarwal, V. Khullar, N. Goyal, A. Alammari, M. A. Albahar, and A. Singh, "Lightweight federated learning for rice leaf disease classification using non independent and identically distributed images," *Sustainability*, vol. 15, no. 16, p. 12149, 2023.

[9] A. Chakrabarty, S. T. Ahmed, M. F. U. Islam, S. M. Aziz, and S. S. Maidin, "An interpretable fusion model integrating lightweight CNN and transformer architectures for rice leaf disease identification," *Ecol. Inform.*, vol. 82, p. 102718, 2024.

[10] M. Y. Alhasnawi, A. A. J. Al-Hchaimi, Y. R. Muhsen, and A. Lekmiti, "A Bibliometric Review of Trends and Insights of Internet of Things on Cybersecurity Issues BT - Current and Future Trends on AI Applications: Volume 1," M. A. Al-Sharafi, M. Al-Emran, M. A. Mahmoud, and I. Arpaci, Eds. Cham: Springer Nature Switzerland, 2025, pp. 127–147. doi: 10.1007/978-3-031-75091-5_8.

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

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

[13] A. A. J. Al-Hchaimi, A. H. M. Alaidi, Y. R. Muhsen, M. F. Alomari, N. Bin Sulaiman, and M. U. Romdhini, "Optimizing Energy and QoS in VANETs through Approximate Computation on Heterogeneous MPSoC," in *2024 4th International Conference on Emerging Smart Technologies and Applications (eSmarTA)*, 2024, pp. 1–6.

[14] G. Latif, S. E. Abdelhamid, R. E. Mallouhy, J. Alghazo, and Z. A. Kazimi, "Deep learning utilization in agriculture: Detection of rice plant diseases using an improved CNN model," *Plants*, vol. 11, no. 17, p. 2230, 2022.

[15] P. Dhiman *et al.*, "A novel deep learning model for detection of severity level of the disease in citrus fruits," *Electronics*, vol. 11, no. 3, p. 495, 2022.

[16] J. Chen, D. Zhang, Y. A. Nanehkaran, and D. Li, "Detection of rice plant diseases based on deep transfer learning," *J. Sci. Food Agric.*, vol. 100, no. 7, pp. 3246–3256, 2020.

[17] J. Chen, W. Chen, A. Zeb, S. Yang, and D. Zhang, "Lightweight inception networks for the recognition and detection of rice plant diseases," *IEEE Sens. J.*, vol. 22, no. 14, pp. 14628–14638, 2022.

[18] R. Gajjar, N. Gajjar, V. J. Thakor, N. P. Patel, and S. Ruparelia, "Real-time detection and identification of plant leaf diseases using convolutional neural networks on an embedded platform," *Vis. Comput.*, pp. 1–16, 2022.

[19] M. Kumar, S. Gupta, X.-Z. Gao, and A. Singh, "Plant species recognition using morphological features and adaptive boosting methodology," *IEEE Access*, vol. 7, pp. 163912–163918, 2019.

[20] A. L.-S. B. Dataset, "Plant Disease Recognition: A Large-Scale Benchmark Dataset and a Visual Region and Loss Reweighting Approach".

[21] A. Chaudhury and J. L. Barron, "Plant species identification from occluded leaf images," *IEEE/ACM Trans. Comput. Biol. Bioinforma.*, vol. 17, no. 3, pp. 1042–1055, 2018.

[22] Y. Wang, H. Wang, and Z. Peng, "Rice diseases detection and classification using attention based neural network and bayesian optimization," *Expert Syst. Appl.*, vol. 178, p. 114770, 2021.

[23] K. Mahadevan, A. Punitha, and J. Suresh, "A Novel Rice Plant Leaf Diseases Detection Using Deep Spectral Generative Adversarial Neural Network,"

Int. J. Cogn. Comput. Eng., 2024.

[24] H. B. Prajapati, J. P. Shah, and V. K. Dabhi, "Detection and classification of rice plant diseases," *Intell. Decis. Technol.*, vol. 11, no. 3, pp. 357–373, 2017.

[25] S. M. T. Islam, M. A. Masud, M. A. U. Rahaman, and M. M. H. Rabbi, "Plant leaf disease detection using mean value of pixels and canny edge detector," in *2019 International Conference on Sustainable Technologies for Industry 4.0 (STI)*, 2019, pp. 1–6.

[26] A. Islam, R. Islam, S. M. R. Haque, S. M. M. Islam, and M. A. I. Khan, "Rice leaf disease recognition using local threshold based segmentation and deep CNN," *Int. J. Intell. Syst. Appl*, vol. 13, no. 5, pp. 35–45, 2021.

[27] K. Anandhan and A. S. Singh, "Detection of paddy crops diseases and early diagnosis using faster regional convolutional neural networks," in *2021 international conference on advance computing and innovative technologies in engineering (ICACITE)*, 2021, pp. 898–902.

[28] M. Li *et al.*, "High-performance plant pest and disease detection based on model ensemble with inception module and cluster algorithm," *Plants*, vol. 12, no. 1, p. 200, 2023.

[29] A. Kaur, K. Guleria, and N. K. Trivedi, "A deep learning-based model for biotic rice leaf disease detection," *Multimed. Tools Appl.*, pp. 1–27, 2024.

[30] A. L. A. Haikal, N. Yudistira, and A. Ridok, "Comprehensive Mixed-Based Data Augmentation For Detection of Rice Leaf Disease in The Wild," *Crop Prot.*, p. 106816, 2024.