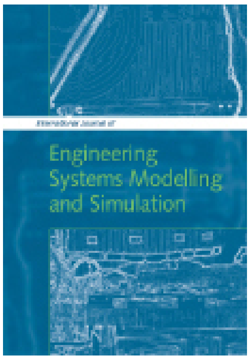
INDERSCIENCE PUBLISHERS
Linking academia, business and industry through research



International Journal of Engineering Systems Modelling and Simulation

ISSN online: 1755-9766 - ISSN print: 1755-9758
https://www.inderscience.com/ijesms

Rice plant diseases detection using convolutional neural networks

Manoj Agrawal, Shweta Agrawal

DOI: 10.1504/IJESMS.2022.10044308

Article History:

Received:	25 July 2021
Accepted:	02 September 2021
Published online:	03 December 2022

Rice plant diseases detection using convolutional neural networks

Manoj Agrawal*

Faculty of Engineering and Technology,
Sage University,
Indore, Madhya Pradesh 452020, India
Email: manoj2179@gmail.com
*Corresponding author

Shweta Agrawal

Department of Computer Science and Engineering,
Sage University,
Indore, Madhya Pradesh 452020, India
Email: shweta.sagecse@gmail.com

Abstract: Rice is one of the main crops grown in India and it is complicated for farmers to accurately classify rice diseases manually with their imperfect information. Thus, the automatic recognition of rice plant diseases is highly desired. Many methods are available and have been proposed for the rice plant diseases detection. The latest advances indicate that the use of CNN models can be very beneficial in such troubles. In this paper we have explored and trained various CNN models with the unique combinations of training and learning methods to enhance the accuracy. The most advanced large-scale architecture, such as VGG19, XceptionNet, ResNet50, DenseNet, SqueezeNet, and CNN are implemented with the baseline and transfer learning methods. These models are trained and tested on datasets collected from various sources. Experimental results show that the ResNet50 architecture achieved the highest accuracy of 97.5% as compared to other CNN architectures and existing literature.

Keywords: convolutional neural network; CNN; deep learning; base learning and transfer learning; rice leaf diseases; VGG19; XceptionNet; ResNet50; DenseNet; SqueezeNet.

Reference to this paper should be made as follows: Agrawal, M. and Agrawal, S. (2023) 'Rice plant diseases detection using convolutional neural networks', *Int. J. Engineering Systems Modelling and Simulation*, Vol. 14, No. 1, pp.30–42.

Biographical notes: Manoj Agrawal is currently a research scholar at the Faculty of Engineering and Technology, Sage University, Indore, Madhya Pradesh, India. He is having more than 18 years of experience involving teaching, research and other academic works. He received his Master's in Computer Applications (MCA) from Rajiv Gandhi Technical University, Bhopal in 2003 and MTech in Computer Science and Engineering from Mewar University, Chittorgarh, India in 2016. He has published more than 20 research papers in various international journals and conferences. His research interest includes machine learning, deep learning, neural networks and artificial intelligence.

Shweta Agrawal is a Professor in Computer Science and Engineering Department at SAGE University, Indore. She has received Best Teaching Award and Work of Excellence Award at university level. She received her MTech in Computer Science and Engineering in 2010 and PhD in Computer Science and Electronics from Devi Ahilya University in 2015. She has qualified UGC-NET in 2013. She has coordinated and received funding for many national and state level FDPs and conferences. She has published four national patents and more than 30 research articles. Her research interests are robotics, controls, distributed computing, machine learning and deep learning.

1 Introduction

In the current circumstance, the total populace keeps on rising quickly, while the developed land appropriate for development remains something similar. It powers farmers to show inventive strategies that further develop crop yields to support the expanding populace. Safeguarding the strength of yields is very basic in this regard. In this manner, early recognition of disease in crops is by all accounts exceptionally significant to screen illness and increment crop yield.

Plant health is closely related to food safety. The FAO (2017) concludes that the loss of global crops caused by pests or diseases is in between 20% to 40% in world food production and represents a threat to food sanctuary (FAO, 2017). The use of pesticides is one way to protect crops from these violations or thus maintain yields. Though, the use of such a substance is not harmless to the environment. The application of these substances can have a negative impact on biodiversity, including insects, birds and fish, as well as on the quality of soil, air and water (Knillmann and Liess, 2019). Its use also represents a risk to human health, with acute and chronic effects (Kim et al., 2017). Though the number of bug juice used worldwide is escalating or active ingredients used increased by 78% between 1990 and 2016 (FAO, 2018).

2 Related work

Early recognition of sicknesses in plants is perhaps the most difficult issues in farming. In the event that the diseases are not recognised in the beginning phases, they may antagonistically influence the total yield, bringing about a reduction in the farmers' benefits. To beat this issue, numerous analysts have introduced distinctive cutting edge frameworks dependent on deep learning and machine learning advances.

Iwendi et al. (2021) carries out an experimental analysis to find out the usefulness and overall performance of deep learning algorithms in indentifying affronts in social media comments. The deep learning models that have been used for experimental outcomes were bidirectional long short-term memory (BLSTM), gated recurrent units (GRU), long short-term memory (LSTM), and recurrent neural network (RNN). The results demonstrate that the BLSTM model achieved high accuracy and F1-measure scores in comparison to RNN, LSTM, and GRU.

Chittathuru et al. (2021) examines ways to recognise malnutrition affected individuals and fat people by analysing BMI and body weight from facial images by proposing a regression method supported on the 50-layers residual network architecture. For face recognition, multi-task cascaded convolutional neural networks (CNNs) have been employed. A system is built to measure BMI in conjunction with age and gender from human facial real-time images.

Five dissimilar tea tree diseases were identified from leaves. A neural network was trained using climatic parameters such as temperature, relative humidity, rainfall, or wind speed to predict the explosion of rice. Hughes and Salathé (2016) used deep CNN to detect leaf diseases, using 54,306 images of 14 crops instead of 26 diseases, while Sladojevic et al. (2016) used the ResNet model to identify 13 diverse types of plant diseases, Wang et al. (2017) used the plant village dataset to study the four stages of severity of apple black rot. They used CNN architectures with diverse depths or execute two different training techniques in each of the architecture. Kim et al. (2017) used deep learning to build a real-time tomato plant disease detector. Brahimi et al. (2017) used optimised AlexNet and GoogleNet to perceive nine tomato diseases. Cruz et al. (2017) insert some characteristics of texture or shape into a fully connected layer located after the convolutional layer, so that model can effectively detect Olive's rapid decline syndrome from a set of limited data. Wiesner-Hanks et al. (2017) did not scale the image down or train model end-to-end, instead assume a three-stage architecture (consisting of multiple CNNs) and the stage was qualified to build a model in a full-size image by separating a single image into several smaller images. Barbedo et al. (2018) uses transfer learning (TL) on GoogleNet to detect 56 diseases that infect 12 plants. Ferentinos (2018) used a data set of 87,848 images of leaves imprison in the laboratory or in the field to study 58 types of plants, including 25 different plants. Reputable CNN that combined ideas of AlexNet and GoogLeNet to detect four apple diseases uses images of personality lesions or spots rather than images of whole leaves to identify 79 diseases of 14 plants. There are few studies on the classification of rice diseases (Liu et al., 2017) behaviour and learning that detect ten unusual diseases of rice plants using a small manual architecture from CNN, which was inspired by 500 images, from previous deep learning frameworks (such as LeNet-5 or AlexNet). He used AlexNet (large-scale architecture) to use 227 images to distinguish three types of normal rice plants, diseased rice plants, and snail-infested rice plants. Determining the healthiness of plants through images is a very complex task. Its evolution is constant, with leaves, flowers, or fruits that change throughout the season. Its emergence also modifies slightly during the day, because the amount of angle of incident solar radiation affects its spectral response. Either under controlled conditions or real conditions, a variety of techniques has been used to develop techniques for the identification of crop diseases. These technologies are especially based on the analysis of the reflectance of visible and near-infrared light, or the progress of specific vegetation indices even through analysis of models. These studies also found some issues that make it difficult to effectively use these technologies in the diagnosis of spontaneous diseases. Some of these problems are solvable and are connected to image availability, weather restrictions, application costs, availability, speed, or actual diagnostic capabilities. Analysing images in the field adds other issues, such as the ability to deal with complex elements such as leaves or their different origins. Other bottlenecks are associated with the severity of isolation problems, such as changes in symptoms over time and type, or the risk of multiple illnesses at the same time. Technology that can overcome these challenges is needed to produce automated disease diagnosis solutions. Deep learning architectures especially the data centre network (CNN), is highly effective in various computer vision tasks such as object detection or recognition, organisation and biometric recognition. The opposite layer of CNN can be viewed as a corresponding filter derived from the data. Thus, CNN can create a level of visual descriptions for specific jobs. The result of CNN training is to obtain a model, a series of weights or biases, or then to respond to the specific tasks targeted by the model. One of CNN's major advantages is its versatility, that is, the ability to organise data that has never been noticed before. This makes it resistant to background heterogeneity, image status or intra-class variability. However, the study of these products requires a great amount of training data. Given the multiple scales, a common problem with DNNs is their tendency to forget training data, which means they become free to select appropriate architecture for a particular difficulty or interpreting training results (produced by the black box) are other challenges facing CNN. For more information about CNN, Picon et al. (2019) studied three CNN architectures that combine metadata (such as product information) into an image-based filter network. This mingles the benefits of learning from multiple pieces of data while reducing the difficulty of disease organisation tasks. The park and disease classification system optimises all of the above methods. It eliminates 71% of the above methods by combining the facts' data at the vector level to reach a value of 0.98, not a category. By using CNN's growth characteristics, we get a richer or more robust visual image, with an average BAC of 0.98, which is better than the other method, or 71% of the time. Classification errors are removed. This demonstrates that other metadata can be easily incorporated into the in-depth learning model to achieve classification criteria that are superior to other methodological classification data. The proposed method uses all static information and data variables set without being affected by similar symptoms in plants. Sharma et al. (2020) work investigates possible solutions to this problem by training a CNN model using segmented image data. Compared to the F-CNN model trained with the full image, the S-CNN model trained with the segmented image doubled its performance when tested with independent data that even ten disease category models could not see before. 98.6% accurate. Ramamurthy et al. (2020) investigate the plant village data set, which contains three ailments, namely early blight, late blight, and leaf mould. The planned work uses the attention mechanism, uses the characteristics learned by CNN in different processing hierarchies, and achieves an in general precision of 98% in justification set in a five-time cross-validation. Shunmugam and Dharmar (2020) proposed the use of an optimised deep neural network and Jaya's algorithm to identify and classify rice leaf diseases. For image collection, images of normal rice plants with bacterial diseases, brown spots, pod rot and rice blight were captured directly from the farmland. In pre-processing, to remove background, the RGB image is transformed to an HSV image, or binary image is extracted according to hue or saturation part to segment diseased and non-diseased parts.

The main constraint of two-stage instruction is that the entire data set must be manually divided into symptom categories. In large data sets, detecting all the important differences within a class is a very labour-intensive process. Some changes in symptoms are very likely to be missed. Minor changes in a particular category can be misread as disconnect symptoms. One probable solution is to use high-dimensional clustering algorithms on category-specific image sets to automate the procedure of recognising changes within the category. Confusion matrix produces by applying simple CNN on the whole data set (combining training or validation sets). 4.3% of the false smut images present in the data set is misclassified, maximum among all existing categories in this work. Compared to other existing images of pests and diseases, symptoms of false blight cover a small part of the whole image (captured on a heterogeneous background). The left and right columns respectively symbolise the production of the first stage or second stage of the simple CNN model. Each of the six images contains 16 two-dimensional micro-images with a size of 222×222 (output of first convolutional layer is a matrix with a size of $222 \times 222 \times 16$). The output of the last convolutional layer of simple CNN has been produced with similar configurations. Each of these six images encloses 64 two-dimensional thumbnail images with a size of 10×10 (the last convolutional layer generates a matrix with a size of $10 \times 10 \times 64$). Although some filters are blank (inactive), the first layer preserves the regional characteristics of the input image. Trigger retains almost all information present in the input image. The output of the final convolutional layer is visually difficult to understand. This depiction shows less information about the visual content of the input image. The transitional result of different categories is visually dissimilar for unusual categories. Compared to the second-stage model, the output of the final convolutional layer of the first stage model carries a much smaller number of blank two-dimensional thumbnail images. This shows that the second stage model has less efficient learning capacity. This helps simple CNN to get good precision and high precision after the second stage of training. Plant diseases are an important issue in agricultural production and if they cannot be discovered in time, they will have a negative impact on the yield and quality of crops as we all know, early detection and early warning are the foundation for effective prevention and control of plant diseases and plant diseases play an important role in management and decision-making. Many previous works have addressed this problem by processing plant leaf images and developing special categories to classify the samples. The reason plant leaf imagery was chosen as the analytical data is that plant leaf is often the first area of occurrence of most plant diseases (Garcia and Barbedo, 2016). With the help of computer science and technology (Garcia and Barbedo, 2019), there are two methods: conventional machine learning and deep learning. Common machine learning algorithms used for disease recognition include K-nearest neighbours (KNN) (Singh and Kaur, 2018), support vector machines (SVM) (Naik and Sivappagari, 2016), rainforests (RF) (Chaudhary et al., 2016). However, these traditional methods have had a significant impact on the shape of the hand in various ways, such as histogram oriented gradient (HOG), scale invariant feature transform (SIFT), Gabor transformation, principal component analysis (PCA), etc. In-depth research techniques, particularly the network of genetic engineers (CNN), have gained considerable attention when creating agricultural images such as plant diseases and insects (Kamilaris and Prenafeta-Boldú, 2018) and (Li and Chao, 2020; Chen et al., 2020; Li and Yang, 2020; Thenmozhi and Reddy, 2019). CNN's model dominates the field of image processing research (Too et al., 2019). This phenomenon has to be linked to CNN's powerful investigative work, which can be applied immediately through the hair-raising layer. As we all know, a high-quality CNN-related layer can produce important features, such as edges, textures and colours. As the CNN layer deepens, the actions taken will become more abstract. Therefore, it is necessary to see if CNN depth is needed to diagnose plant diseases, or if CNN depth can provide sufficient form to deal with it. This algorithm is applied to high-end devices (such as GPUs and servers). To get better performance, the depth of the network will be deepened with multiple computational constraints, which consume a lot of resources and time. However, due to insufficient production capacity, high-quality materials may be unsuitable for field applications. In contrast, embedded software has better software for simplicity and cost (Castro et al., 2020). In addition, considering the satisfaction, low power consumption and low computational cost, the application of high-end mobile devices that maintain intelligent classification algorithms is a calming trend (Tao et al., 2020). Based on the above analysis, we hope to explore CNN's ability to resolve the symptoms on the ground. In this article, we propose two methods to treat nuclear disease: SCNN-KSVM (CNN nonsense with SVM kernel) and SCNN-RF (CNN nonsense with sudden forest). The features of the plant images can be obtained immediately from nearby CNN, and then combined into traditional machine learning algorithms, such as the SVM kernel and abrupt forest When comparing machine learning classification algorithms, the SVM algorithm is an important classification algorithm that can avoid excessive reactions, and it is also one of the most widely used machine learning classification algorithms (Liakos et al., 2018). Meanwhile, the forest dense algorithm has very good efficiency and speed (Liang et al., 2020). Therefore, SVM seeds and unselected forests were selected to classify the works produced by CNN. Our method is compared to other in-depth learning models with three different data sets. The facts show that our method is superior to other deep learning models in terms of accuracy, memory and F1 numbers, while (Qi et al., 2020) have a lower score. When the duration of period data is short, the performance of the traditional data obtained from the SMF method will decrease. In contrast, the estimated duration of the proposed network is well consistent with the results of the soil survey, the primary level is 83.9%, and the total error (MAE) is 0.18. For the estimated phonological period, the division of the space at harvest time of 627 plots in the study area was calculated. The predicted value corresponds well to the date of the harvest found. The results show that the in-depth analysis method has good success in determining the actual physical time and evaluating the harvest time.

This article focuses on the accurate definition and classification of rice diseases. For this reason, various CNN architectures have been implemented, such as VGG19, XceptionNet, ResNet50, DenseNet, SqueezeNet and CNN were used.

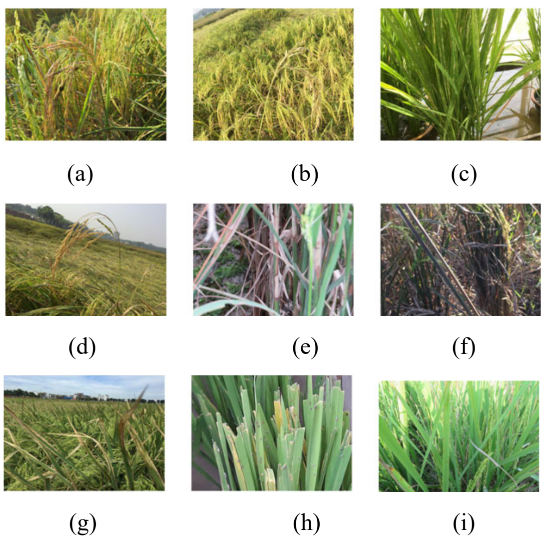
- The major contribution of this work is to develop a machine learning architecture that can accurately (with > 95% accuracy) identify and classify rice disease.
- Comparative analysis of deep learning models that are commonly used for the detection of rice diseases.
- Development of CNN based disease detection and classification model based on available datasets
- Improvisation of the developed model to predict pest attacks with high level of precision.

3 Different types of rice plant diseases

- *Leaf blast (LB):* symptoms of this disease are dark spots with reddish-brown round spots and white or white or white centres (Kim et al., 2017).
- *Brown spot (BS):* this disease occurs on rice leaves. The symptoms of the disease are round to oblong, and the lesions are yellowish (Kim et al., 2017).
- *Sheath blight (SB):* this disease occurs on leaves and stems. Symptoms are an elongated, white or straw-like area in the centre with a reddish-yellow dot (Kim et al., 2017).
- *Leaf scald (LS):* the symptom is a narrow reddish-brown band. Sometimes the lesions are on the margins of the leaves, the margins yellow or gold (Kim et al., 2017).
- *Bacterial leaf blight (BLB):* the lesions appear as thin, long leaf tips, which are several centimetres long, and the bacterial reaction varies from white to yellow (Kim et al., 2017).
- *Rice blast (RB):* it is because of fungus Magnaporthe Orysa. White to gray-green lesions or spots, with dark green borders in an initial stage. More established injuries on the leaves are elliptical or spindle-shaped and whitish to gray centres with red to a brownish or necrotic border. Spots are usually lengthened and pointed at each end (Brahimi et al., 2017).
- *Sheath rot (SR):* it is created by two fungal species, Sarocladium Orysa and Sacroladium attenuate. The typical Sheath rot injury starts at the uppermost leaf sheath encasing the little panicles. It seems oblong or as an asymmetrical spot with dark reddish, brown margins, and gray centres or brownish-gray all through. Usually, several spots are observed and these spots enlarge and merge or rise together and can cover up most of the leaf sheath. Panicles remain within the sheath or may partially emerge. Affected leaflets may have plentiful whitish powdery fungal growth

(mycelium) visible on the outer surface. Panicles that have not appeared rot and the florets turn red-brown to dark brown (Brahimi et al., 2017). Figure 1 images showing symptoms of various rice plant diseases and pests (Agrawal and Agrawal, 2020). In some images, the background is the surroundings of the field, and in some other images, the background is our hand or papers of different colours. Weather conditions are also different at different times. Some images have been captured in overcast conditions; some have been captured in sunny weather. False smut, stem borer, healthy plant class, sheath blight and/or sheath rot class have multiple types of symptoms. This proposed covered all the symptoms of these classes. Moreover, early-stage symptoms of Hispa and brown plant hopper are different from their later stage symptoms.

Figure 1 Detected class of rice plant diseases, (a) bacterial leaf blight (disease) (b) brown plant hopper (pest) (c) brown spot (disease) (d) false smut (disease) (e) stemborer (pest) (f) Hispa (pest) (g) neck blast (disease) (h) sheath blight (disease) (i) sheath rot (see online version for colours)



4 Problem statement

New technologies now playing a huge role in helping the farmers elevate their rice production. Farmers are looking into a faster and more reliable solution, a solution where they can easily take feasible action with their diseased rice crops. Most of the current plant leaf disease detection techniques are either classifying if the plant is diseased or they identify the diseased area of the leaf. The data provided is not big enough to allow the usage of a deep neural network which would provide a more accurate classification. The data set is considered very small, and images of different rice diseases look very similar, deep neural networks cannot be trained from scratch since the data is very small, shallow feature extraction produced around 80%. The challenge here is to use machine learning techniques, pre-trained deep learning models exist which can produce generic features for models such as VGG19, XceptionNet, ResNet50, DenseNet, SqueezeNet and CNN to not only classify if the rice plant is diseased or not but to also classify which disease it is and provide more accurate results.

5 Methodology

5.1 Rice plant disease detection by well-known deep learning architectures

After the introduction of AlexNet (Castro et al., 2020) for image recognition, segmentation or classification, many next-generation DL models/architectures have been developed. This section presents an investigation on the classification of plant diseases using well-known DL architecture. Also, there are some connected works where new visualisation practices or modified/enhanced versions of DL architecture have been initiated to achieve better results. Among all of these, the plant village dataset has been widely used because it includes 54,306 images of 14 dissimilar crops suffering from rice plant diseases. Additionally, they used various concert metrics to appraise the preferred DL model, as explained below.

- *Convolutional layer:* the first layer of the CNN architecture is called the convolutional layer. It is used to learn the detection filters for basic features such as edges, corners, etc. It is the number of filters with a specific size that will convolve the input image.
- *Pooling/sub-sampling layer:* the middle layer is the grouping layer, also called the sub-sampling layer. This layer helps to learn the filter that detects the part of the object. It helps to reduce the size of feature map space and is used to reduce the number of parameters. They perform three types of functions, namely, maximum group, average group, and sum group:
- *Max pooling:* this function takes the maximum value in a given filter area.
- *Average pooling:* this function gets the average value in the filter area.
- *Sum pooling:* this function summarises all the elements of the feature map.
- *Fully connected layer:* the last layer in the architecture is called the fully connected layer. This layer has a higher representation. This helps to learn to recognise whole objects of different shapes and positions. It takes the results of the convolutional layer and the grouping layer, processes them, and uses them to classify the image to be labelled.
- *SqueezeNet architecture:* this architecture was proposed by Iandola et al. (2016). He uses three main strategies to improve the effectiveness of the traditional CNN network. First, most of the filters used in a network are 1×1 rather than 3×3; this greatly reduces the number of network values. Second, it reduces the

number of channels entering the 3 × 3 filter. This method also significantly reduces the number of network weights. Third, create a network block afterward on a larger activation map. The proposed hypothesis is that there is a direct relationship between the size of the activation map by down sampling and the accurate final classification results. Another important strategy to significantly reduce the number of network weights is to remove the commonly used thick layer in the last layer of the network and replace it with a convolutional layer with the same number of output channels. By the number of data classes, followed by the flight and softmax activation function.

- *ResNet-50:* in the last two CNNs, what we have seen an increase in the number of layers in a form and achieving better performance. But as the depth of the network increases, the accuracy rate gradually increases, and then decreases rapidly. People at Microsoft Research have used ResNet to solve this problem by tracing the connections (also known as short, residuals), and in the meantime building a more in-depth model. The ResNet architecture is one of the first contributors to batch normalisation. A deeper CNN mixed with more layers will address the problem of loss of gradients. To solve this problem, an additional model that has been trained with additional layers is used to implement the identity map. The efficiency of deeper networks and lighter networks should be the same. The remaining study structure was proposed as a solution to the problem of impairment. Therefore, the remaining map ($H(x) = F(x) + x$) instead of the required base plate ($H(x)$) is inserted into the network and is called a model.
- *VGG-19:* the visual geometry group network (VGGNet) is a deep-seated network with multiple operating layers. VGGNet is based on the CNN model and is applied to ImageNet advertising (Singh and Kaur, 2018). VGG-19 is necessary for its simplicity, as a 3 × 3 convolutional layer is placed on top of it to increase its depth. To reduce the size, the topmost collection layer is used as the processing program in VGG-19. Two fully connected layers are applied to 4,096 neurons. In the training process, the convolutional layer is used for feature extraction, and the topmost delineation layer is associated with some convolutional layers to reduce refinement of features. In the first Convolutional layer, 64 kernels (3 × 3 filter size) are applied to remove shapes from the input image. A fully integrated layer is used to prepare the featured content.
- *Xception:* dedicated to himself by François Cholet, the creator and chief maintainer of the Keras library. Xception is an extension of the Inception architecture. It replaces the traditional Inception model with a convolution that can be viewed separately. Xception has the smallest serialisation value, only 91MB.
- *DenseNet:* DenseNet is one of the discoveries in neural networks for visual object recognition. DenseNet is quite similar to ResNet with some fundamental differences. ResNet uses an additive method (+) that merges the previous layer (identity) with the future layer, whereas DenseNet concatenates (.) the output of the previous layer. n output of the previous layer acts as an input of the second layer by using composite function operation. This composite operation consists of the convolution layer, pooling layer, batch normalisation, and non-linear activation layer. These connections mean that the network has $L(L + 1)/2$ direct connections. L is the number of layers in the architecture. The DenseNet has different versions, like DenseNet-121, DenseNet-160, DenseNet-201, etc. The numbers denote the number of layers in the neural network.

5.2 Training and evaluation phases

If time or computing property allows, using the same hyperparameters for multiple training sessions can improve accuracy, as random initialisation can affect results. When comparing hyperparameters, it is recommended to consider fixed random number generators to avoid skewing the comparison, which is also desirable. Testing more than one nature of architecture can also play an optimistic role. To obtain the same precision, it is more advantageous to choose the least complicated architecture from a prepared point of view. If related, TL is suggested to recover computational time or generalise ability. After correcting all hyperparameters, the model must be retrained by combining the images previously used for preparation or validation into the overall training set. In fact, previously all hyperparameters are defined; it is no longer necessary to maintain the validation set. So it’s worth using this global training set to try to recover exactness one last time (i.e., no post-adjustment to any hyperparameters). The retrained model can be evaluated on test apparatus. The visualisation step is also imperative because it helps to better recognise what is happening in the model or to ensure the toughness of the results. This advance can also supply opportunities to recover routine.

- *Basic training:* in this process, we train all the layers from scratch. We initialise all the layers at random or train them from graze. This training method takes a long time to converge but produces pretty good accuracy. We denote this preparation process as BT.
- *Fine tuning:* fine-tuning is a way of utilising TL. Specifically, it is a process that takes a model that has already been trained for one given task and then tunes the model to make it perform a second similar task. In this training method, we keep the net weight of pertained image of the convolutional layer unchanged. We just aimlessly initialise weights of compactly related layers. Then we train all the layers to a junction. It should be noted that the convolutional layer is trained

based on the net weights of the pertained image, while the dense layer is trained based on the arbitrarily initialised weights. We denote this technique as FT.

- *Transfer learning:* TL occurs when we use knowledge that was gained from solving one problem and apply it to a new but related problem. For example, knowledge gained from learning to recognise one class of vehicle could be applied in a problem of recognising another class of vehicle with similar features. In this method, we do not train the convolutional layer of the CNN architecture at all. Instead, we retain the previously trained image network weights. We only train dense layers from erratically initialised weights. We denote this process as TL.

5.3 Dataset collection

The rice plant disease image data collection was obtained from various sources. All images are marked with categories or saved in JPG format. The background condition of the image is difficult and the intensity of the illumination is uneven. For further calculations, first use Photoshop tools to render these images into RGB models, and then adjust the size of the images. These images of rice disease mainly include rice pile burns, rice leaf burns, rice leaf blight, white rice sprouts, and rice leaf bacterial stripes. The general process of our rice disease detection method is as follows: First, collect and label image samples of rice disease to the knowledge of experts in the field; then perform image resizing, image sharpening and image edge fill on the acquired images. Internal image dispensation knowledge and the use of rotation and translation data enhancement methods has been used to generate new sample images to enrich data set to generate the expanded data set; then the sample image is entered into the proposed model training method.

Rice pests or diseases occur in unusual parts of rice plants. Its incident depends on many features, such as warmth, humidity, rainfall, rice varieties, seasons, nutrition, etc., and then the task of collecting data at the field level is long or arduous task.

Classifications considered: we have a total of ten rice disease ailment classifications. Symptoms of dissimilar diseases can be observed in dissimilar parts of the rice plant, such as leaves, stems, or grains. There is bacterial leaf blight, brown spot, sheath blight rot1, sheath blight rot2, sheath blight rot3, stemborer, false smut1, neck blast1 and healthy class. We have measured all these parts when incarcerating the image. 1,584 images were taken for the bacterial blight and 1,308 images were taken from the Tungro class (Sethy et al., 2020). To prevent the model from mystifying dead and diseased parts of rice plants, we together enough images of dead leaves, stems, and grains of rice plants. In the category of plant health, images of dead spots of plants are considered. We consider nine classes in total. Example images for each category are provided in Figure 1. Images have been collected from (Chowdhury et al., 2020) research paper, the author collected datasets in a real-life scenario with heterogeneous backgrounds from December 2017 to June 2018 for a total of seven months. The image collection has been performed in a range of weather conditions – in winter, in summer and in the overcast condition to get as fully representative a set of images as possible. Four different types of cameras have been used by (Chowdhury et al., 2020) in capturing the images. There are nine different classes of diseases with one class of healthy images. The class names along with the number of images collected for each class are shown in Table 1. Note that sheath blight, sheath rot and their simultaneous occurrence have been considered in the same class because their treatment method and place of occurrence are the same.

5.4 Proposed approach

We use some modern CNN architectures, such as VGG19, XceptionNet, ResNet50, DenseNet, Mobile SqueezeNet and CNN. The simple CNN is a CNN using a 3 × 3 convolution filter. After each maxpool layer, the number of difficulty filters is doubled in VGG19. In each initial block, convolution filters of various sizes and groupings are used in parallel input. Before providing the output, connect them in series across their channels. In addition to the direct connections from the top layer, you also have jump connections from top layer. XceptionNet, SqueezeNet combines the parallelism concept of the Inception architecture with the jump connection of the DenseNet architecture, and each layer connects directly to other layers in the form of feedback (within each dense block). For each layer, feature maps from all previous layers are considered disconnect inputs and their characteristic maps are passed as the input from all following layers. In addition, we use different learning and training methods for each of the architecture.

Table 1 Image collection of different classes

Class name	*No. of collected images*
Bacterial blight	1,584
Bacterial leaf blight (BLB)	138
Brown spot	111
False smut	93
Healthy	180
Hispa	73
Neck blast	286
Sheath blight 1, 2, 3	219
Stemborer	201
Tungro	1,308

6 Experimental evaluation

We have qualified our dataset using the latest five CNN architectures. They are – VGG19, XceptionNet, ResNet50, DenseNet, and SqueezeNet. We have used baseline learning and TL with these CNN architectures. To evaluate our method with existing methods, we train the model for the data set. If the image size is set to 224 × 224, the precision of the test set is%. The image size declares is 512 × 512. This image size can provide less than 30% verification and proof accuracy. Here, we can see that when we adjust the previously trained ImageNet weights, all of our architectures give the best precision in the test set. On other hand, we did not train various layers of innovative convolutional planning in TL. Therefore, the model may not be able to capture all individuality of the data set. To get the best presentation changes from each architecture (that is when we adjust the architecture), the verification precision and the test precision are very high. Rice images are used which are taken from (Chowdhury et al., 2020) and some of the images are collected from the Kaggle dataset.

Stochastic gradient descent with momentum (SGDM), adaptive moment estimation (ADAM) and root mean square propagation (RMS propagation) techniques are used for training. Performances are examined for the study.

SGDM and its variants are the optimisation method of choice for many large-scale learning problems including deep learning (Barbedo et al., 2018). Momentum is a method that helps accelerate SGDM in the relevant direction and dampens oscillations (Ramamurthy et al., 2020). The momentum term is set to 0.9. As a result, momentum obtains faster convergence and reduced oscillation.

ADAM is an optimisation algorithm. It can be used instead of stochastic gradient descent to update network weights (Sladojevic et al., 2016). This technique computes adaptive learning rates for each parameter. Additionally, ADAM keeps an exponentially decaying average of past gradients similar to momentum (Ramamurthy et al., 2020). ADAM is a popular algorithm in the field of deep learning because it achieves good results quickly (Sladojevic et al., 2016).

RMS Propagation obtains parameter learning rates that are adapted based on the average of recent magnitudes of the gradients for the weight. This means the algorithm is successful for online and non-stationary problems (Sladojevic et al., 2016). RMS propagation divides the learning rate by an exponentially decaying average of squared gradients (Ramamurthy et al., 2020).

6.1 Experimental setup

All experimental studies were conducted on a 64-bit machine with Windows 10 operating system running on Intel i5 processor @ 2.20GHz and 16 GB RAM with 500 GB SSD. MATLAB 2019B was used for coding. Pre-trained models were used from deep learning library.

Table 2 shows the number of parameters used for each of the five architectures alongside simple CNN architecture.

Table 2 Number of parameters in state-of-the-art CNN architectures

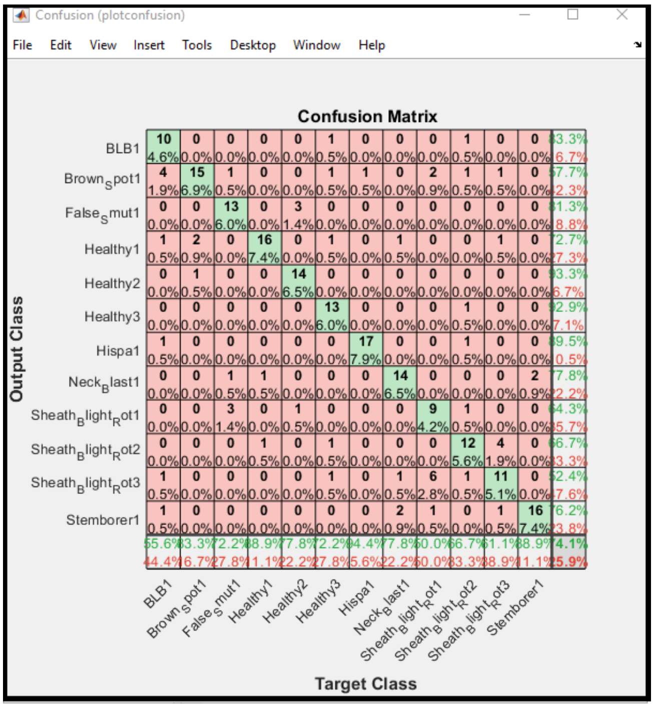
CNN architecture	*Number of parameters*
VGG19	144 million
SqueezeNet	5 million
ResNet50	23 million
DenseNet	20 million
XceptionNet	60 million
Simple CNN	10 million

Our work aims to use deep CNNs to efficiently perceive eight types of pests and diseases that influence rice and healthy rice. An overview of our organisation is shown in Figure 2. We collected a large dataset from nature by capturing images of infected rice plants. We have ten categories: nine disease categories and one healthy plant category. We annotate images by placing images from different categories in separate folders to train our CNN. We aimlessly sample 70% of all images from each class and include them in the training set. Similarly, the other 15% of the images in each category are placed in the verification set and the remaining ones are placed in the test set. We then use different intensity conversion or image enhancement techniques to increase the number of images in the training set tenfold. The model in Table 1 shows the number of images in each category in the training set, validation set, or test set. All images are sized 224 × 224 pixels. We determine the CNN architecture performance evaluation indicators from the overall structure and identified the hyperparameters, adjust these hyperparameters, or set the situation for the research. Figure 2 showing the confusion matrix for the entire dataset used for training.

We have trained our dataset using next-generation CNN architectures. They are VGG19, XceptionNet, ResNet50, DenseNet, SqueezeNet and simple CNN to compare our method with the existing recognition methods of (Chowdhury et al., 2020).

After training of each model we get the classification layer output for each model like weighted 1,000 * 4,096 fully connected layer output getting in VGG-19 model, weighted 1 * 1 * 1,000 fully connected layer output getting in SqueezeNet model. Weighted 1 * 1 * 1,920 fully connected layer output getting in DenseNet model. Weighted 1,000 * 2,048 fully connected layer output getting in InceptionNet model. Weighted 1,000 * 2,048 fully connected layer output getting in ResNet model. Weighted 1,000 * 4,096 fully connected layer output getting in InceptionNet model. They are simple CNN models. Each of the six images contains two 16-dimensional thumbnails with a size of 222 × 222 (the end of the first coordinate field is a matrix of the same size 222 × 222 × 16). The six images contain 64 double-dimensional images with a size of 10 × 10 (the output of the last convolutional layer is a matrix with a size of 10 × 10 × 64).

Figure 2 Confusion matrix of dataset (see online version for colours)



The first layer retains the peripheral properties of the input image, although there are some filters. Activation stores almost all of the information found in the introductory image. The exit of the last convolutional layer is not easy to understand. This representation defines a lack of visible internal information about the introductory image. Instead, this layer attempts to display information related to the image category. For different classes, the median results with different classes differ in visibility. The release of the final proposal of the first model yields a slightly less mixed image than the second phase model which demonstrates the ability to learn fewer features. This helps CNN achieve high accuracy and precision after the second phase of training. To improve the ability of deep neural networks, the most direct way is to increase the depth of the network. However, as the depth of the network width increases, there are too many internal dimensions, which results in more resource consumption. Therefore, in order to overcome these problems, Sladojevic et al. (2016) introduced the Inception model into the GoogleNet architecture and then completed an impressive performance and read the record as the winner of the ImageNet ILSVRC Challenge. The first model consists of an upper layer and a corresponding plate. The sizes of the mixed layers were 1 × 1, 3 × 3 and 5 × 5, which were combined. Between two 1 × 1 separate layers, a max-pooling layer is used to reduce the dimensionality, and a concatenation filter is required to mix the different layers. DenseNet connects the outputs of all layers to the barriers that all layers insert into it. The thick barrier is the repetition of batch normalisation, ReLU, 1 × 1 convolution, batch normalisation, ReLU, and 3 × 3 convolution over a period of time, as we see the three layers block. Each time after the thick barrier, the translation layer reduces the size as a 1 × 1.

7 Result and discussion

Process performance is precisely based on concert indicators such as precision, sensitivity, specificity, or time consumption. Performance measure: in our data set, the sample size is fairly isolated among 11 categories. When the sample size is not biased towards any particular category, precision is a good performance indicator.

- *TP* – is the total number of properly categorised prospects (true positives).
- *TN* – is the total number of poorly classified prospects (true negative numbers).
- *FN* – is the total number of false rejections, which represents the number of false pixels of foreground pixels classified as background (false negatives).
- *FP* – is the total number of false positives, which means that pixels are mistakenly classified as foreground (false positives).
- *Accuracy:* in the field of material retrieval, accuracy is the number of recovered documents related to the query. A test technique is supposed to be precise when it estimates what it supposed to estimate. It is the ratio of correctly labelled predictions to the total number of predictions.

  $$\text{Accuracy} = \text{Correct number of predictions} / \text{total number of predictions}$$

  $$\text{Accuracy} = (\text{TP} + \text{TN}) / (\text{TP} + \text{FP} + \text{TN} + \text{FN})$$

- *Specificity:* specificity is distinct as a proportion of definite refusals that can be predicted as negatives (or true negatives).

  $$Specificity = (TP) / (TN + FP)$$

- Precision: precision is used with the retrieval rate, which is the percentage of all relevant documents returned by the search. A test technique is supposed to be precise when repeated determinations (analyses) on the same sample give similar results.

  $$Precision = (TP) / (TP + FP)$$

- *Recall (sensitivity):* recall is the ratio of correctly positive identified subjects by test against all positive subjects in reality.

  $$Recall = TP / (TP + FP)$$

- *F1 score:* F1 score considers both precision and recall.

  $$F1\ Score = 2 * (Recall * Precision) / (Recall * Precision)$$

Figure 3 shows the interface for learning and training method used.

Figure 4 shows the interface for different CNN models used for rice disease detection.

Figure 5 shows the input images and Figure 6 shows the corresponding segmentation images.

Figure 3 Learning and training method (see online version for colours)

Table 3 showing the quantitative performance of the different CNN models with different learning algorithms and training methods. In case of the VGG-19 model for baseline learning method with SGDM training method the accuracy has come 90%, for ADAM, accuracy has come 92%, for RMS propagation accuracy has come 85%. In case of the TL method with SGDM training method the accuracy has come 90%, for ADAM accuracy has come 92%, and for RMS propagation accuracy has come 97.45%.

Figure 4 Implemented CNN models (see online version for colours)

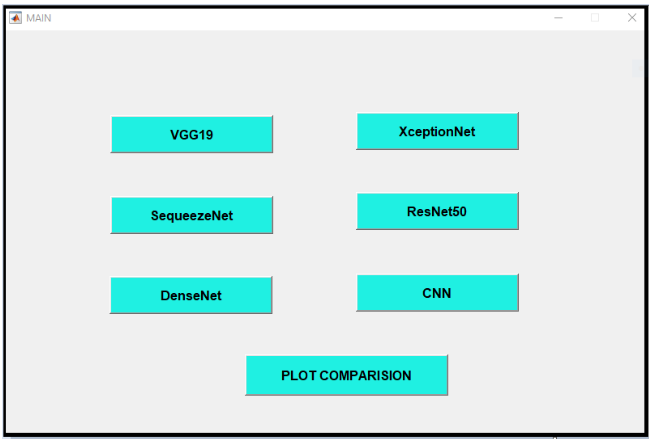


Figure 5 Input images (see online version for colours)

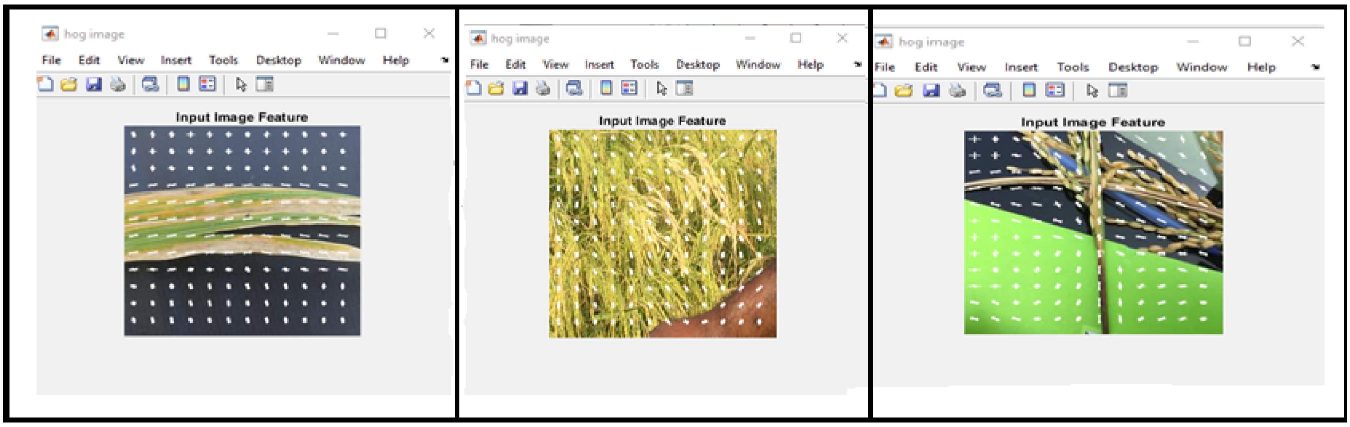


Figure 6 Segmentation images (see online version for colours)

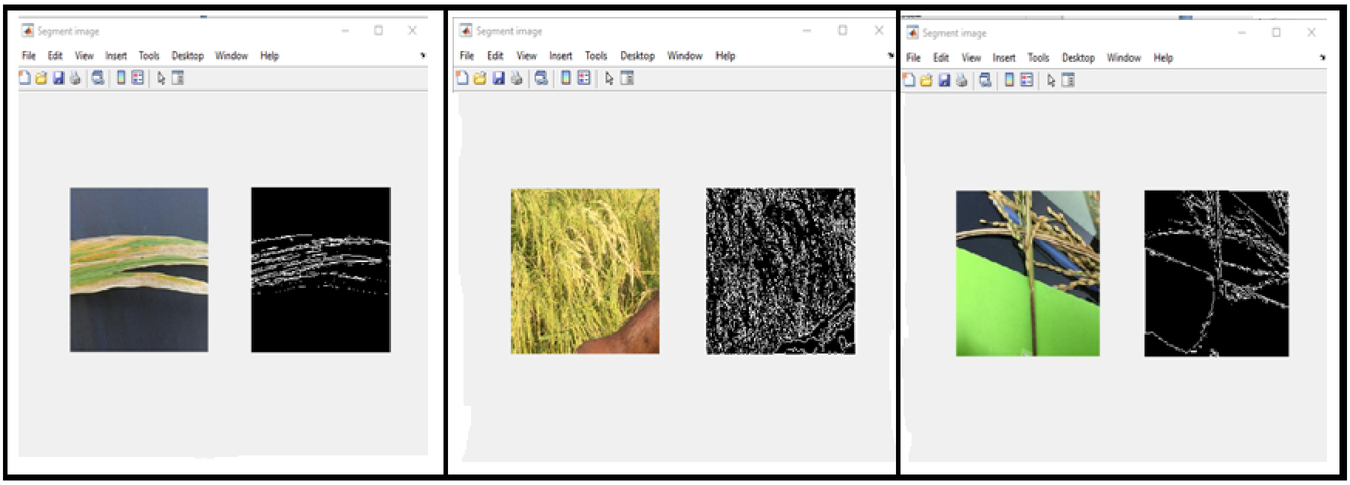


Table 3 Quantitative performance of different CNN architectures

Deep learning architecture	*Learning method*	*Training method*	*Validation accuracy*
VGG 19	Baseline	SGDM	90%
		ADAM	92%
		RMS propagation	85%
	Transfer learning	SGDM	90%
		ADAM	92%
		RMS propagation	*97.25%*
SqueezeNet	Baseline	SGDM	95%
		ADAM	89%
		RMS propagation	92%
	Transfer learning	SGDM	89%
		ADAM	85%
		RMS propagation	*96%*
ResNet50	Baseline	SGDM	92%
		ADAM	93%
		RMS propagation	94%
	Transfer learning	SGDM	96%
		ADAM	*97.50%*
		RMS propagation	94%
DenseNet	Baseline	SGDM	92%
		ADAM	91%
		RMS propagation	90%
	Transfer learning	SGDM	90%
		ADAM	*94.50%*
		RMS propagation	86%
XceptionNet	Baseline	SGDM	93%
		ADAM	94%
		RMS propagation	93%
	Transfer learning	SGDM	91%
		ADAM	*96.50%*
		RMS propagation	93%
CNN	Baseline	SGDM	93%
		ADAM	91%
		RMS propagation	86%
	Transfer learning	SGDM	91%
		ADAM	86%
		RMS propagation	*95%*

In the case of the SqueezeNet model, the accuracy for the baseline learning method with SGDM training method the accuracy has come to 95%, for ADAM the accuracy has come 89%, for RMS propagation the accuracy has come 92%. In the case of TL method with the SGDM training method the accuracy has come to 89%, for ADAM the accuracy has come 85%, and for RMS propagation the accuracy has come 96%.

Table 4 Comparison of proposed model with existing techniques

Reference	*Year*	*Deep learning architecture*	*Learning/ training method*	*Accuracy*
Hu et al.	2019	SVM	Deep learning	90%
Garcia and Barbedo	2019	CNN	Transfer learning	85%
Picon et al.	2019	CNN	Learning algorithm	93%
Yang et al.	2020	CNN	SGDM	83.9%
Sambasivam and Opiyo	2020	CNN	Deep learning	93%
Rahman et al.	2020	Deep learning models	Baseline/ transfer learning	95%
Proposed techniques		VGG-19	Transfer learning/RMS	97.25%
		SquuezeNet	Transfer learning/RMS	96%
		ResNet50	Transfer learning/ADAM	97.50%
		DenseNet	Transfer learning/ADAM	94.50%
		XceptionNet	Transfer learning/ADAM	96.50%
		CNN	Transfer learning/RMS	95%

In case of the ResNet50 model, the accuracy for baseline learning method with SGDM training method has come to 92%, for ADAM the accuracy has come 93% and for RMS propagation the accuracy has come 94%. In case of TL method with SGDM training method the accuracy has come 96%, for ADAM the accuracy has come 97% and for RMS Propagation the accuracy has come 94%.

In case of the DenseNet model the accuracy for baseline learning method with SGDM training method has come 92%, for ADAM the accuracy has come 91% and for RMS propagation the accuracy has come 90%. In case of TL method with SGDM training method the accuracy has come 90%, for ADAM the accuracy has come 94.5% and for the RMS propagation the accuracy has come 86%.

In the case of XceptionNet model, the accuracy for the baseline learning method with SDGM training method has come 93%, for ADAM the accuracy has come 94% and for RMS propagation the accuracy has come 93%. In case of TL method with SGDM training method the accuracy has come 91%, for ADAM the accuracy has come 96.5% and for RMS propagation the accuracy has come 93%.

In case of the CNN model the accuracy for baseline learning method with SGDM training method has come 93%, for ADAM the accuracy has come 91% and for RMS propagation the accuracy has come 86%. In case of TL method with SGDM training method the accuracy has come 91%, for ADAM the accuracy has come 86% and for RMS propagation the accuracy has come 95%.

In Table 3, for each of the architecture, the best accuracy achieved has been mentioned in bold character.

Table 4 showing the quantitative performance of proposed architectures with the previous work done by different authors. The proposed models are compared with the other deep learning model. After the training and testing process, we have found that the ResNet50 model gets higher accuracy as compared to other models.

8 Conclusions

We have proposed a rice disease detection classifier based on deep CNNs. We have conducted an extensive investigation on rice plant diseases which includes nine types of rice diseases, pests and healthy plants. We have used a dataset of different rice plant diseases consisting of 4,193 images. We apply agricultural knowledge to solve the problem of the classification of rice diseases. We use various types of CNN architectures and implemented the unique combination of various training and learning methods in each of the architectures. In this paper, we evaluated the performance of five different CNN architectures namely VGG-19, SqueezeNet, ResNet50, DenseNet, XceptionNet and simple CNN. TL and baseline methods have been implemented on these architectures on the dataset collected from various sources. The experimental result shows that the ResNet50 model achieved the highest accuracy as compared to other CNN architectures and existing literature. We have been able to successfully discriminate among within-class variations of diseases and rice plants in complex environments. The verification precision and test precision of most CNN architectures are very high because of our training, verification and testing. We plan to combine location, climate and soil data, or images of diseased plant parts to develop a complete or automated plant disease detection mechanism. Due to a large number of parameters, our CNN architecture is very large. We plan to make efforts to achieve a high-efficiency classification accuracy of plant diseases and insect pests and deploy them with efficient CNN architecture. Our model is 98.2% accurate on independent test images. Also, due to the reduced number of network parameters, our model is efficient for memory storage. Although the precision is higher, our goal is to recover the reliability or robustness of the model in dissimilar data sets from other regions. When the background is complex and the lighting conditions change, we will classify the picture of rice leaf disease because categorisation accuracy is an imperfect explanation of most real-world tasks. In future, we intend to deploy the hybrid approach for the two or more models in which we have obtained the highest accuracy. One can also implement these architectures with lesser number of training parameters. Moreover a large dataset can also be used to achieve higher accuracy. This will help developing models that can make more accurate predictions in difficult environments.

References

Agrawal, M. and Agrawal, S. (2020) ‘Rice plant diseases detection & classification using deep learning models: a systematic review’, *Journal of Critical Reviews*, Vol. 7, No. 11, pp.4376–4390.

Barbedo, J.G.A. (2018) ‘Factors influencing the use of deep learning for plant disease recognition’, *Biosystems Engineering*, Vol. 172, No. 1, pp.84–91, Elsevier Publications.

Brahimi, M., Boukhalfa, K. and Moussaoui, A. (2017) ‘Deep learning for tomato diseases: classification and symptoms visualization’, *Applied Artificial Intelligence*, Vol. 31, No. 4, pp.299–315, Elsevier Publications.

Castro, A., Madalozzo, G.A., Trentin, N.S. et al. (2020) ‘BerryIP embedded: an embedded vision system for strawberry crop’, *Computers & Electronics in Agriculture*, Article 105354, Vol. 173, No. 1, pp.1–13, Elsevier Publications.

Chaudhary, A., Kolhe, S. and Kamal, R. (2016) ‘An improved random forest classifier for multi-class classification’, *Information Processing in Agriculture*, Vol. 3, No. 4, pp.215–222, Elsevier Publications.

Chen, J., Chen, J., Zhang, D. et al. (2020) ‘Using deep transfer learning for image-based plant disease identification’, *Computers & Electronics in Agriculture*, Vol. 173, No. 6, pp.1689–1699, Elsevier Publications.

Chittathuru, D., Un, N., Reddy, P.K., Gadekallu, T.R., Iwendi, C. et al. (2021) ‘Identification of malnutrition and prediction of BMI from facial images using real time image processing and machine learning’, *IET Image Processing*, Vol. 15, No. 5, pp.1–12, John Wiley & Sons Ltd., DOI: 10.1049/ipr2.12222.

Cruz, A.C., Luvisi, A., De Bellis, L. and Ampatzidis, Y. (2017) ‘X-FIDO: an effective application for detecting olive quick decline syndrome with deep learning and data fusion’, *Frontiers in Plant Science*, Article 1741, Vol. 8, No. 10, pp.1–8.

Ferentinos, K.P. (2018) ‘Deep learning models for plant disease detection and diagnosis’, *Computers & Electronics in Agriculture*, Vol. 145, No. 2, pp.311–318, Elsevier Publications.

Food and Agriculture Organization of the United Nation (FAO) (2018) *FAOSTAT Pesticides Use Dataset* [online] http://www.fao.org/faostat/en/#data/RP (accessed 4 April 2019).

Food and Agriculture Organization of the United Nation, International Plant Protection Convention (FAO) (2017) *Plant Health and Food Security*, https://www.fao.org/3/I9003EN/i9003en.pdf (accessed 4 April 2019).

Garcia, J. and Barbedo, A. (2016) ‘A review on the main challenges in automatic plant disease identification based on visible range images’, *Biosystems Engineering*, Vol. 144, No. 4, pp.52–60, Elsevier Publications.

Garcia, J. and Barbedo, A. (2019) ‘Plant disease identification from individual lesions and spots using deep learning’, *Biosystems Engineering*, Vol. 180, No. 4, pp.96–107, Elsevier Publications.

Hu, G., Wu, H., Zhang, Y. and Wan, M., (2019) ‘A low shot learning method for tea leaf's disease identification’, *Computers & Electronics in Agriculture*, Elsevier Publications, Vol. 163, No. 8, pp.181–186.

Hughes, D.P. and Salathé, M. (2016) *An Open Access Repository of Images on Plant Health to Enable the Development of Mobile Disease Diagnostics through Machine Learning and Crowd Sourcing*, ArXiv, abs/1511.08060.

Iandola, F.N., Han, S., Moskewicz, M.W., Ashraf, K., Dally, W.J. and Keutzer, K. (2016) 'SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and < 0.5 mb model size', in *International Conference on Learning Representations (ICLR)*, Vol. 2, pp.236–248.

Iwendi, C., Srivastava, G., Khan, S. et al. (2021) 'Cyberbullying detection solutions based on deep learning architectures', *Multimedia Systems*, Vol. 26, No. 5, pp.607–620, Springer publications.

Kamilaris, A. and Prenafeta-Boldú, F. (2018) 'Deep learning in agriculture: a survey', *Computers & Electronics in Agriculture*, Vol. 147, No. 4, pp.70–90, Elsevier Publications.

Kim, K-H, Kabir, E. and Jahan, S.A. (2017) 'Exposure to pesticides and the associated human health effects', *The Science of the Total Environment*, Vol. 575, No. 1, pp.525–535.

Knillmann, S. and Liess, M. (2019) 'Pesticide effects on stream ecosystems', in Schröter, M., Bonn, A., Klotz, S., Seppelt, R. and Baessler, C. (Eds.): *Atlas of Ecosystem Services*, Springer International Publishing AG, Part of Springer Nature, Cham, https://doi.org/10.1007/978-3-319-96229-0_33.

Li, Y. and Chao, X. (2020) 'ANN-based continual classification in agriculture', *Agriculture*, Vol. 10, No. 5, pp.178–192.

Li, Y. and Yang, J. (2020) 'Few-shot cotton pest recognition and terminal realization', *Computers & Electronics in Agriculture*, Article 105240, Vol. 169, No. 2, pp.1–9, Elsevier Publications.

Liakos, K.G., Busato, P., Moshou, D. et al. (2018) 'Machine learning in agriculture: a review', *Sensors*, Article 2674, Vol. 18, No. 8, pp.1–29, Elsevier Publications.

Liang, X.W., Jiang, A.P., Li, T. et al., (2020) 'LR-SMOTE – an improved unbalanced data set oversampling based on K-means and SVM', *Knowledge Based System*, Article 105845, Vol. 196, No. 5, pp.1–10, Elsevier Publications.

Liu, B., Zhang, Y., He, D. and Li, Y. (2017) 'Identification of apple leaf diseases based on deep convolutional neural networks', *Symmetry*, Vol. 10, No. 11, pp.1–16, Elsevier Publications.

Naik, M. and Sivappagari, C. (2016) 'Plant leaf and disease detection by using HSV features and SVM classifier', *International Journal of Engineering, Science & Computing*, Vol. 6, No. 12, pp.3794–3797, Elsevier Publications.

Picon, A., Seitzc, M. et al. (2019) 'Crop conditional convolutional neural networks for massive multi-crop plant disease classification over cell phone acquired images taken on real field conditions', *Computers & Electronics in Agriculture*, Vol. 167, No. 12, pp.1–16, Elsevier Publications.

Rahman, C.R. et al. (2020) 'Identification and recognition of rice diseases and pests using deep convolutional neural networks', *Biosystems Engineering*, Vol. 194, No. 6, pp.112–120, Elsevier Ltd.

Ramamurthy, K., Hariharan, M., Anand S., Mathikshara, P. et al. (2020) 'Attention embedded residual CNN for disease detection in tomato leaves', *Applied Soft Computing*, Vol. 86, No. 1, pp.1–27, Elsevier Publications.

Sambasivam, G. and Opiyo, G.D. (2020) 'A predictive machine learning application in agriculture: cassava disease detection and classification with imbalanced dataset using convolutional neural networks', *Egyptian Informatics Journal*, Vol. 22, No. 1, pp.27–34.

Sethy, P.K., Barpanda, N.K., Rath, A.K. and Behera, S.K. (2020) 'Deep feature based rice leaf disease identification using support vector machine', *Computers and Electronics in Agriculture*, Article 105527, Vol. 175, No. 8, pp.1–9, Elsevier Publications.

Sharma, P., Berwal, Y. and Ghai, W. (2020) 'Performance analysis of deep learning CNN models for disease detection in plants using image segmentations', *Information Processing in Agriculture*, Vol. 7, No. 4, pp.566–574, Elsevier Publications.

Shunmugam, R. and Dharmar, V. (2020) 'Recognition and classification of paddy leaf diseases using optimized deep neural network with Jaya algorithm', *Information Processing in Agriculture*, Vol. 7, No. 2, pp.249–260, Elsevier Publications.

Singh, J. and Kaur, H. (2018) 'Plant disease detection based on region-based segmentation and KNN classifier', in *International Conference on ISMAC in Computational Vision and Bio-Engineering*, Vol. 30, No. 1, pp.1667–1675, Springer.

Sladojevic, S., Arsenovic, M., Anderla, A., Culibrk, D. and Stefanovic, D. (2016) 'Deep neural networks based recognition of plant diseases by leaf image classification', *Computational Intelligence & Neuroscience*, Article ID: 3289801, Vol. 2016, No. 6, pp.1–11.

Tao, M., Ma, X., Huang, X. et al. (2020) 'Smartphone-based detection of leaf color levels in rice plants', *Computers & Electronics in Agriculture*, Article 105431, Vol. 173, No. 6, pp.1–10, Elsevier Publications.

Thenmozhi, K. and Reddy, S. (2019) 'Crop pest classification based on deep convolutional neural network and transfer learning', *Computers & Electronics in Agriculture*, Article 104906, Vol. 164, No. 9, pp.1–11, Elsevier Publications.

Too, E., Yujian, L., Njuki, S. et al. (2019) 'A comparative study of fine-tuning deep learning models for plant disease identification', *Computers & Electronics in Agriculture*, Vol. 161, No. 6, pp.272–279, Elsevier Publications.

Wang, G., Sun, Y. and Wang, J. (2017) 'Automatic image-based plant disease severity estimation using deep learning', *Computational Intelligence & Neuroscience*, Article ID: 2917536, Vol. 2017, No. 7, pp.1–8.

Wiesner-Hanks, T., Stewart, E.L., Kaczmar, N., DeChant, C., Wu, H., Nelson, R.J. et al. (2018) 'Image set for deep learning: field images of maize annotated with disease symptoms', *BMC Research Notes*, *Part of Springer Nature*, Article 440, Vol. 11, No. 1, pp.1–3.

Yang, Q., Shi, L., Han, J., Yu, J. and Huang, K. (2020) 'A near real-time deep learning approach for detecting rice phenology based on UAV images', *Agricultural & Forest Meteorology*, Article 107938, Vol. 287, No. 6, pp.1–17, Elsevier Publications.