BETEL LEAF DISEASE RECOGNITION USING DEEP LEARNING

BY

RASHIDUL HASAN HRIDOY
ID: 171-15-8596

This Report Presented in Partial Fulfillment of the Requirements for the Degree of Bachelor of Science in Computer Science and Engineering.

Supervised By

MD. TAREK HABIB
Assistant Professor
Department of CSE
Daffodil International University

DAFFODIL INTERNATIONAL UNIVERSITY

DHAKA, BANGLADESH

JANUARY 2021

APPROVAL

This Project titled "**Betel Leaf Disease Recognition Using Deep Learning**", submitted by Rashidul Hasan Hridoy, ID: 171-15-8596 to the Department of Computer Science and Engineering, Daffodil International University, has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of B.Sc. in Computer Science and Engineering and approved as to its style and contents. The presentation has been held on 31 January 2021.

BOARD OF EXAMINERS

____

Dr. Touhid Bhuiyan
Professor and Head **Chairman**
Department of CSE
Faculty of Science & Information Technology
Daffodil International University

____

Moushumi Zaman Bonny
Assistant Professor **Internal Examiner**
Department of CSE
Faculty of Science & Information Technology
Daffodil International University

____

Dr. Md Arshad Ali **External Examiner**
Associate Professor
Department of Computer Science and Engineering
Hajee Mohammad Danesh Science and Technology University

DECLARATION

We hereby declare that, this project has been done by us under the supervision of **Md. Tarek Habib, Assistant Professor, Department of CSE,** Daffodil International University. We also declare that neither this project nor any part of this project has been submitted elsewhere for award of any degree or diploma.

Supervised by:

____

Md. Tarek Habib
Assistant Professor
Department of CSE
Daffodil International University

Co-Supervised by:

____

Mr. Sheikh Abujar
Lecturer (Senior Scale)
Department of CSE
Daffodil International University

Submitted by:

Rashidul Hasan Hridoy

____

Rashidul Hasan Hridoy
ID: 171-15-8596
Department of CSE
Daffodil International University

ACKNOWLEDGEMENT

First we express our heartiest thanks and gratefulness to almighty **Allah** for His divine blessing makes us possible to complete the final year project/internship successfully.

We really grateful and wish our profound our indebtedness to **Md. Tarek Habib**, Assistant Professor, Department of CSE Daffodil International University, Dhaka. Deep Knowledge & keen interest of our supervisor in the field of "*Machine Learning*" to carry out this project. His endless patience, scholarly guidance, continual encouragement, constant and energetic supervision, constructive criticism, valuable advice, reading many inferior draft and correcting them at all stage have made it possible to complete this project.

We would like to express our heartiest gratitude to **Dr. Touhid Bhuiyan**, Head**,** Department of CSE, for his kind help to finish our project and also to other faculty member and the staff of CSE department of Daffodil International University.

We would like to thank our entire course mate in Daffodil International University, who took part in this discuss while completing the course work.

Finally, we must acknowledge with due respect the constant support and patients of our parents.

ABSTRACT

In South Asia, a large number of people consume betel life every day and it becomes a part of the daily life of this reason for many years. Betel leaf is also used as a remedy and, it's a great source of nutrition. In the international market demands for betel leaf is increasing incredibly. Plants of betel leaf are highly sensitive, so its cultivation is very laborious work. Like other South Asian countries, farmers of Bangladesh also face a great loss for the diseases of betel leaf every year. Diseases of betel plants decrease leaves quality. Recognition of betel leaf diseases became a crucial task to meet the rising demand in local and international markets and increase the quality of leaf. We will recognize the diseases of betel leaf using deep learning. First, we study some online articles, journals, and, related papers then we talk to local farmers and visit their lands; we find two common diseases of betel leaf which are foot rot and, leaf rot. Then we collect images of healthy and affected betel leaves. We also collected some images of unknown diseases. After data collecting images, we resized images and create a dataset of four classes. We applied deep learning models in our dataset to recognize the disease of betel leaf. Now deep learning is used in various recognition related systems. We have proposed a CNN model named BLCNN, which achieved 90.75% training accuracy and 89.44% test accuracy. Besides BLCNN we have also modified 3 pre-trained models for this study. Among these, EfficientNet B0 has performed better than other models. It has achieved 96.77% accuracy in classifying new images.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGRES

CHAPTER 1

Introduction

1.1 Introduction

The popularity of deep learning is increasing now incredibly, it's a very young field of AI which is inspired by the brain's function and structures, called Artificial Neural Network. It is capable of dealing with huge data that are produced every day. Now computers can understand images because of deep learning algorithms. Various deep learning algorithms are used in different fields to solve real-life problems significantly. To find the best solution for a specific problem, deep learning algorithms need to be trained and tested with data thoroughly.

In South Asia, the Betel leaf is an important part of the culture. It has remarkable usages in the daily life of this region. It also creates a great impact on the economy of this region. Like other South Asian countries, a large number of farmers of Bangladesh also cultivates betel leaf. In Bangladesh, over nine million people work as agricultural laborers and, about 16.5 million families earn their living from agriculture [1]. Bangladesh is also classified among the Next Eleven countries [2]. Its economy largely depends on agriculture and, the demand for agro-processing products is growing incredibly in export markets. In South Asian countries, farmers are unfamiliar with modern technologies. In this region, most of the agriculturalists are uneducated. In Bangladesh, 73.91% of people are educated [3], but most of the farmers are uneducated. They use traditional tools and techniques of agriculture. Every year they face a huge loss due to the lack of knowledge about modern technologies. The disease of plants becomes a matter of deep concern for farmers and governments as the demands for agricultural products increasing rapidly day by day in local and international markets. Farmers can be benefited if they can detect diseases at an early stage. Then they can take proper action to control the diseases. This can be acquired by using an online computer vision-based expert system which will be able to detect diseases of betel leaf from an image taken by a smartphone. Scientific classification of betel plant is given below in Table 1.1.

Table 1.1: Betel scientific classification chart

Scientific Classification	
Kingdom	Plantae
Clade	Tracheophytes
Clade	Angiosperms
Clade	Magnoliids
Order	Piperales
Family	Piperaceae
Genus	Piper
Species	**P. betle**
Binomial name	
Piper betle	

We have used CNN, it is now widely used in various fields for its significant performance than others. In recent times, CNNs are used widely in various leaf disease recognition like cucumber [4], rice [5], and tomato [6].

1.2 Motivation

The economic condition of the South Asian farmers is not as good as other countries. Their life depends fully on agriculture. Like other countries of this region, a large portion of farmers of Bangladesh cultivates betel leaf for the whole year. They borrow money with a high-interest rate for cultivation. Profit from betel leaf cultivation depends on the quality of betel leaf. Diseases of betel leaf is a great threat to farmers. Farmers sell low-quality betel leaf at a cheap rate. Sometimes disease destroys all plants of betel leaf. Sometimes diseases destroy all plants of betel leaf. As a result, farmers face financial loss which makes farmers' life miserable. If they can recognize diseases of betel leaf at an early stage, then they can save them from financial loss. This will also help them to enhance their economic condition. Betel leaf also contains many nutrition such as protein, minerals, fiber, chlorophyll, carbohydrate, nicotinic acid, vitamin C, vitamin A, calcium [7]. It is also used as a remedy for many diseases. To ensure a better quality of betel leaf, recognition of its diseases has become an important task. We are going to recognize diseases of betel leaf using deep learning as we have not found much research work on this field.

1.3 Objectives

In this research work, we are going to a CNN model to recognize betel leaf diseases which will be able to use smart devices. There is less remarkable work that has been done previously in this field, we mentioned that earlier. Moreover, as far we know none use deep learning algorithms. That's why we are interested to work with betel leaf diseases and deep learning techniques. Our expert system will be able to detect disorders from an image taken by handheld technology.

Deep learning is a branch of Artificial intelligence (AI), AI has transformed our world like electricity. Deep learning enables power to researchers to make more exciting and smart AI applications that can autonomously perform tasks without any human intervention. The popularity of CNN is increasing incredibly, CNN is used in now everywhere. It has shown significant performance in the image recognition field. Models built with CNN can be easily used on every platform. As CNN has used in remarkable research work, we thought that we should apply CNN for disease recognition of betel leaf.

1.4 Expected Outcome

We hope this research work will help farmers who are involved with betel leaf cultivation to recognize diseases of betel leaf. Our trained CNN models will be deployed in the smart cloud-based applications. Successful deployment of existing or new deep learning algorithms for recognizing diseases of betel leaf. This helps farmers to ensure the quality of betel leaf easily. This work will help people to know more about recognition using deep learning. New farmers who are unfamiliar with the diseases of betel leaf will also be benefited. If farmers can detect diseases at an early stage, then it will also help to reduce the cost of cultivation. The use of fertilizer in cultivation will also be decreased. This research work will also help to increase the demand for betel leaf in the international market.

The ministry of agriculture of Bangladesh and other countries can make collaborate with our models which will help them to increase the production of betel leaf. Besides, the building of a large image dataset for betel leaf disease in the context of Bangladesh and other South Asian countries. One or more articles will be published in international conference proceedings or journals are based on this work.

1.5 Report Layout

In this report, we proposed a CNN models to recognize the disease of betel leaf, it contains six parts which are given as below:

Chapter 1: Introduction

We explain the introduction of the research work with its motivation, objectives, and expected outcome in this chapter.

Chapter 2: Background Study

We discuss the literature review, research summary, the scope of the problem, and challenges in this chapter.

Chapter 3: Research Methodology

We address the workflow of this research, data collection procedure, implementation requirements in this chapter.

Chapter 4: Experimental Results and Discussion

We cover the result and experimental evaluation, the outcome of research graphically in this chapter.

Chapter 5: Impact on Society, Environment, and Sustainability

We explain how our research impacts society in this chapter.

Chapter 6: Conclusion

This chapter contains a summary of our research along with future work.

CHAPTER 2

Background Study

2.1 Introduction

We will discuss the literature review, research summary, the scope of the problem, and the challenges we faced during this work in this chapter. We summarize some research papers related to our work, we explain underlying methods, classifiers, and accuracies of their works in the part of the literature review. We provide a table in the research summary part in which we display the summary of related works for better understanding easily. We explain how we can contribute to the problem with our CNN models in the scope of the problem part. In the challenges part, we discuss the obstacles we faced during our research work.

2.2 Literature Review

Many research work has been done in recognition related problems of various leaf diseases using the traditional machine learning approach. As far we know none have worked with deep learning to recognize betel vine leaf disease. However, a lot of researchers define their works in recognizing vegetable and fruit diseases, for example, carrots, broccoli, cabbage, junk fruit, mango, dragon, and so on.

Tamilsankar et al. [8] have proposed a computer-aided disease identification approach for betel leaf. They have used a small dataset of 100 images. To remove they have used a median filter. For color transformation, they have used the CIELAB space model. To remove background images they have used watershed segmentation. To obtain gradient feature value histogram, they have used oriented gradients (HOG) techniques. Finally, the minimum distance classifier has been used to identify healthy or affected leaves and the type of their diseases.

To detect disease of betel vine leaf Jayanthi et al. [9] have also used HOG and multiclass SVM (support vector machine) classifiers and their classifier has achieved 95.85% accuracy. For pre-processing of images, they have used a median filter. The watershed algorithm has been used for image segmentation. They have conducted two different models, the color space model and the watershed transformation algorithm. The color space model achieved 82.35% accuracy. To analyze the performance of their work, they have used Sensitivity, specificity, accuracy.

Dey et al. [10] have used only 12 samples. For the segmentation of betel vine leaf, they have implemented the Otsu thresholding-based image processing algorithm. To minimize the noise in the images HSV color space (hue, saturation, value) has been used. To analyze the performance of their classifier, they have used precision and recall. They have used a color feature to recognize affected or rotted leaves from healthy leaves. Their classifier does not show the result correctly as they have used a very small dataset.

Vijayakumar et al. [11] have used ten sample leaves with digital image processing to recognize powdery mildew disease of betel leaf. They have used the mean and median values of images. They have conducted two experiments, in first they have used mean values and the median value has been used in the second experiment. They have used three types of leaves, and these are normal, fully infected, and test leaves.

Whereas, Ramesh et al. [12] have proposed a clustering method to classify betel leaf disease. They have converted RGB image into HSV format. For image segmentation, they have used the *K*-means cluster technique and for feature classification, HSV has been used. In their research work, they have not provided enough information about their experiment result and the performance of their approach.

In another work, Sladojevic et al. [13], have used a deep neural network to recognize plant diseases using the classification of leaf images. OpenCV framework has been used to resize images. They have used the image augmentation technique to get high accuracy by increasing the size of the dataset. In their research work, CaffeNet has been used as a pre-trained model. They have found that in overall accuracy fine-tuning has not shown remarkable changes but the augmentation process has shown notable changes and achieved a high accuracy of 96.3%. Their approach achieved precision between 91% and 98% in separate class tests.

To classify diseases of the leaf, Ishak et al. [14] have used a feed-forward neural network. For the color transformation, they have used HSV. Using color and shape of leaf images they have analyzed leaf diseases. For classification, a Back-propagation algorithm has been used. They have compared the performance of Multi-layer Perceptron (MLP) and Radial Basis Function (RBF) using 100 samples. The classifier of the neural network has achieved more accuracy and remarkable performance. In their research work, MLP achieved 94.05% and RBF achieved 99.1% of classification accuracy using 80 training and 20 test samples.

To detect leaf disease, Sabro et al. [15] have computed texture feature vectors, correlation, contrast, energy, and homogeneity. They have achieved overall recognition accuracy of 90.7% for tomato and 98% for eggplant. For plant disease recognition, they have proposed gray-level spatial dependence based feature extraction. For computing features, a gray level co-occurrence matrix (GLCM) has been used and for disease recognition, an adaptive neuro-fuzzy inference system classifier has been used. They have followed a hybrid method for learning purposes and have stopped execution when the root mean squared error (RMSE) reaches the minimum. In their research work, the GLSMPDR scheme with ANFIS classifier has shown significant recognition.

In yet another work Yadav et al. [16], have used AlexNet for feature extraction. For feature selection and optimization, particle swarm optimization (PSO) has been used. They have extracted 34 optimal features from 100 features of 23 classes using particle swarm optimization. Accuracy, precision, *F1*-score, recall have been used for analyzing the performance of their proposed classifier. SVM has been selected as a final classifier which achieved 97.39% of accuracy after analyzing the performance of four classifiers, XGBoost, SVM, Random Forest, and KNN. They have found that AlexNet training accuracy for 500 and 1000 epochs are the same, 97.33%.

Khan et al. [17] proposed a new pre-processing method, a novel binary segmentation method, and a new feature selection method for the detection and classification of leaf diseased spots and achieved 98.08% accuracy. For leaf disease classification, they have highlighted five stages. They have used VGG-19 and VGG-M for feature extraction based on local entropy, local standard deviation, and local interquartile range method most prominent features have selected. They have analyzed the performance of decision tree, cubic SVM, logistics regression, fine *k*-NN, neural network, ensemble subspace discriminant analysis multi-class SVM (M-SVM) and found that M-SVM has shown the best results. To check the performance of their proposed method, they have tested five affected leaves and achieved 98.08% in 10.52 seconds.

Priyadharshini et al. [18] have proposed a CNN model for leaf disease classification and achieved 97.89% accuracy. For image pre-processing, they have used principal components analysis (PCA) whitening. They have analyzed the performance of different kernel sizes and found that 3×3 has shown better results. In their research work, based on LeNet architecture they have presented a classification method. LeNet deep network has been trained by the gradient-descent algorithm. They have used soft margin loss as a loss function. Different train - test ratio has been analyzed and 80 - 20 has provided better accuracy.

Basavaiah et al. [19] have proposed multiple feature extraction techniques for leaf disease classification. Using decision tree classifier they have achieved 90% accuracy and have achieved 94% accuracy using random forest classifier. Color histograms have been extracted for color features, hu-moments has been extracted shape features, haralick features has been extracted texture features. They have employed texture analysis for classification purposes. Computational time has been reduced but they have achieved better accuracy compared to others.

Goswami et al. [20] have compared different proposed methods for leaf disease classification. To improve accuracy, they have suggested extracting more features and using statistical analysis removing repeating features. They have also found that the back-propagation neural network (BPNN) has achieved better accuracy than others. They have focused on some techniques for preprocessing, segmentation, feature extraction, and classification.

For crop leaf recognition Miaomiao et al. [21] have proposed a series of automatic networks based on binary relevance (BR) multilabel learning algorithm. In their research, 85.28% accuracy has been achieved by BR-CNN based on the light-weight NasNet but BR-CNN based on ResNet50 have achieved the best test accuracy of 86.70%. They have analyzed precision, recall, and *F1*-score to check the effectiveness of their proposed method. Their proposed network has been able to estimate crop disease severity also. They have found that BR-CNNs has performed better than LP-CNNs, MLP-CNNs.

Khamparia et al. [22] have used a small dataset but to make images invariant to any transformations data augmentation has been used. They have minimized error using a backpropagation learning algorithm and to check the effectiveness of their proposed method they have used precision, recall, *F1*-score, and support. They have proposed a hybrid convolutional encoder network using CNNs. They have used different filter sizes. Using 2×2, they have achieved 97.50% accuracy but 100% accuracy has been achieved by 3×3 convolution filter size.

Nidhis et al. [23] have proposed a cluster-based leaf disease classification method, and for image segmentation, *K*-means clustering has been used. To detect diseases in images, they have converted clustered images into binary images, and then point feature matching has been used. They have used the percentage of the affected area to find out the severity of the disease of the leaf. In their research to get clarity on the affected region of the leaf, they have derived the gray image from the RGB image. It also has helped them to extract size, color, proximity, and centroids.

To segment spots in affected images, Chuanlei et al. [24] have used a region growing algorithm (RGA). The correlation-based feature selection method has been used in their research. To recognize leaf disease color, shape, and texture features have been used. To select useful features from images they have used RGA, GA-CFS, and SVM and have found that GA-CFS has shown better performance. With the training set, the recognition rate is above 95% and with the testing set, they have achieved above 93% of accuracy. SVM has been used for disease recognition.

Ramakrishnan et al. [25] have used the backpropagation algorithm to detect and classify leaf disease and have achieved 97.41 % accuracy. To increase the color component of the input RGB images, they have converted RGB images to HSV (Hue saturation value) color images to get. They have masked the green pixels and the color and texture feature has been extracted. Color imagery has been used for classification in their research.

To extract color features, Ratnasari et al. [26] have used L*a*b* color space, and GLCM has been used for texture feature extraction. For classification, they have used an SVM classifier, and with an average 5.73 error severity estimation, they have achieved an accuracy of 80%. They have used energy, correlation, contrast, and homogeneity as GLCM texture features. With a low error severity estimation average their proposed model has resulted in high accuracy in identifying sugarcane leaf disease. They have found that wider lesion results in a higher severity estimation in severity estimation.

Agrawal et al. [27] have proposed used multiclass SVM classifier for detection of leaf disease. For segmentation of the affected area, they have used the *K*-means clustering technique. They have achieved an average of 90% accuracy using features from both L*a*b and HSI color model. Using only the L*a*b model for feature extraction, they have achieved an accuracy of 82.5% for healthy leaves.

2.3 Research Summary

Using data mining, machine learning, and deep learning various work of leaf disease detection, classification and recognition have been already done. The use of machine learning, deep learning has increased in various types of research work. In the literature review part, we have discussed several types of research works. We have found that no research work has been done for betel leaf disease recognition using deep learning. Some researchers have used traditional techniques for betel leaf disease recognition and they have used a few images. As a result, their proposed method is not compatible with recent technologies. We have also found that some researchers have achieved better accuracy using CNN for other leaf diseases.

2.4 Scope of the Problem

A common problem we have found after studying different related research works is huge images cannot proceed using their proposed method for betel vine disease recognition. To get a better model for recognition, we have to train it using a large dataset. They have used very few images for their proposed model. As a result, their proposed system will not work well with new test images. Recently a remarkable part of the researcher uses deep learning for recognition related problem and use a large dataset to train their model. Their model can work with modern technologies, it makes their research meaningful and helpful for ordinary people. Therefore, we decided to a model to leave disease recognition using deep learning. It will be compatible with recent technologies and ordinary people will be benefited from it.

2.5 Challenges

We faced some problems, collecting images of betel leaf became a challenging task for us while doing our research work. During heavy rain, diseases of betel leaf are increased. So, we have collected images over the whole year. We have visited many places and have collected a lot of images. An image of betel leaf field is given below in Figure 2.1.

We have focused on the dataset, as for a better model better dataset is the main ingredient. Symptoms of some diseases of betel leaf are quite the same as each other. It's very hard to separate them. We have to spend a couple of days to create a better dataset.

During the training phase, we also faced some problems. We have used a large dataset and it took a long time to train. We also have changed the architecture of BLCNN many times to achieve high accuracy, which is also a challenging task for us. But finally, we have overcome all challenges and achieved high accuracy. Our CNN models has also shown significant performance with new images. Cultivation land of betel leaf is shown in below in Fig. 2.1.

Fig 2.1: Cultivation land of betel leaf.

CHAPTER 3
Research Methodology

3.1 Introduction

Establish a model that will be able to recognize the diseases of betel leaf correctly with less time is the main goal of this research. Helping people who are involved with betel leaf cultivation is another purpose of this research work. We have used CNN in this research. Besides, we have also used three pre-trained models. Fig. 3.1 shows the entire approach of our research work.

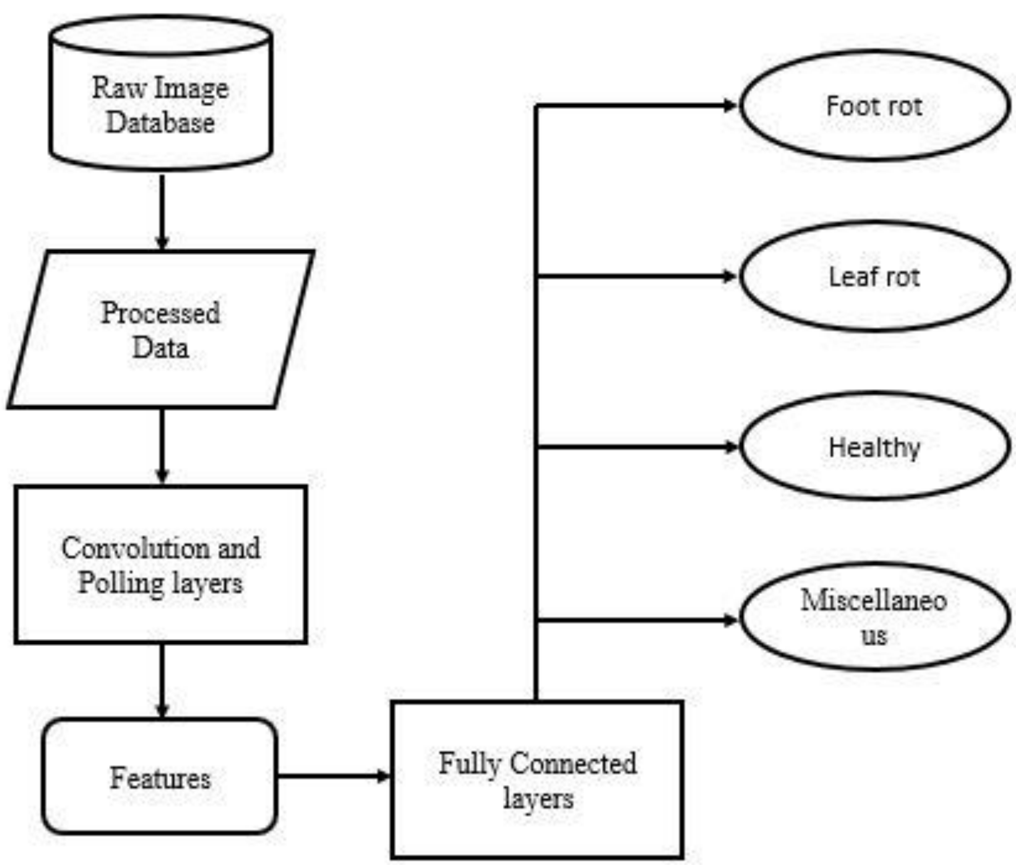

Fig. 3.1: Block diagram of this research methodology

Block diagram illustrates the steps of working for this research work. Dataset is the key ingredient of any research work. Before collecting images, we deeply studied about diseases of betel leaf. After that, we collect images from various places. In this chapter, we discuss about disease description and the data collection procedure briefly. We are going to discuss all the theoretical information in this chapter.

3.2 Disease description

One of the crucial parts of our research work is disease study. It helps us to understand the effects on the leaf for various kinds of diseases of betel leaf easily. After the disease study, we found two common diseases of betel leaf in overall Bangladesh which are leaf rot and foot rot.

Leaf rot is a common disease of betel leaf in Bangladesh as well as South Asia. Another name of leaf rot is anthracnose. Affected seeds spread this disease. In the rainy season, it affects most. It is also known as a leaf spot. It is also spread through the air. High humidity also helps this disease to spread within a short time. In the affected leaves, light to dark brown spots surrounded with diffuse chlorotic yellow halo is shown. Marginal leaf tissue becomes black after some days and it spreads to the center. After some days plat dies as these spots growing rapidly. It's a reason for the great financial loss of farmers.

Another destructive fungal disease is foot rot. During the rainy season, it mostly occurs. To save from a great loss farmers need to burn all affected plants. In the leaf, Foot rot creates wet rot symptoms. The fungus can survive for a long time in affected plants and soil of the land. Quickly it damages the root system and then the plant dies. Within seven days, most of the plants die. In the betel leaf, the dark brown spot appears at the very beginning. The spot becomes wet after some days and then the affected area becomes rot. These two diseases make the farmer's life miserable for the whole year. Most of the farmers don't know well about the diseases, as a result, they can't take any proper action to reduce loss. The images of foot rot, leaf rot, miscellaneous and healthy are shown in below Fig. 3.2.

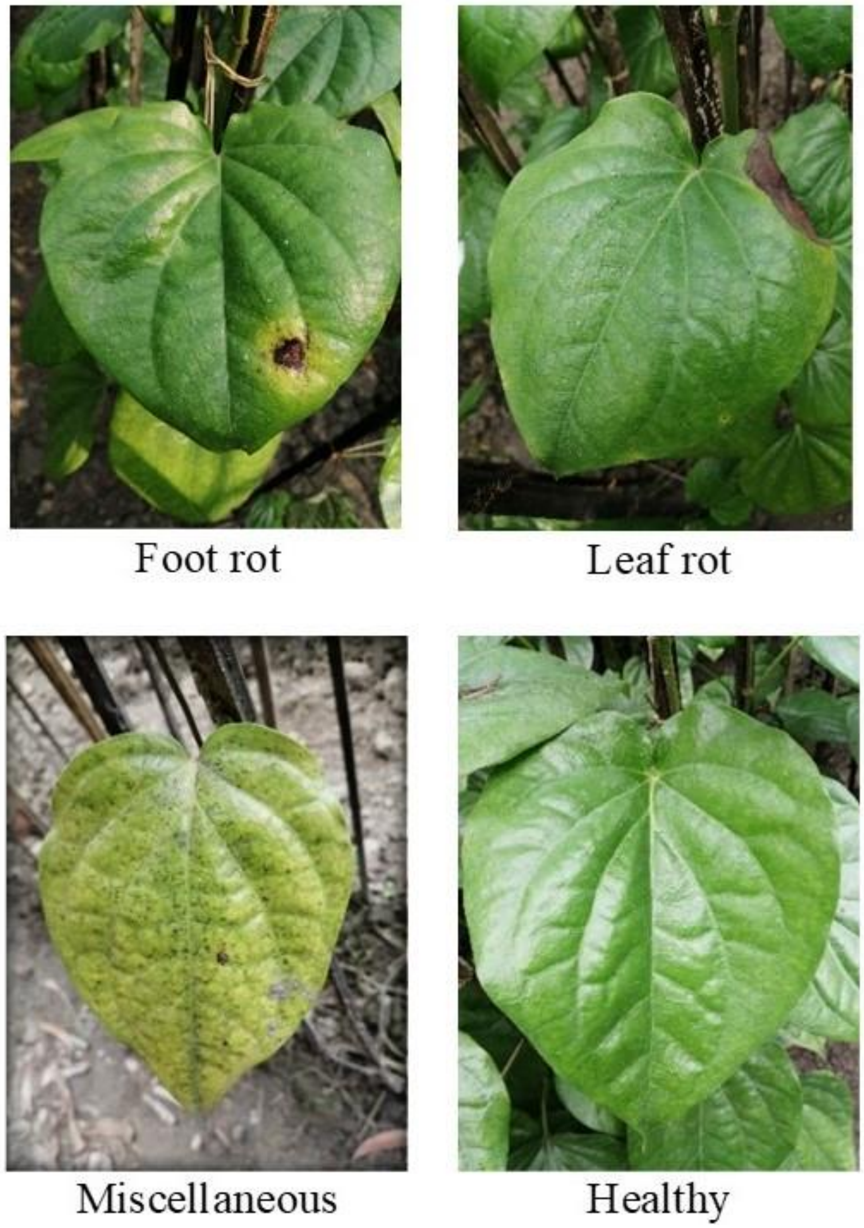

Fig. 3.2: Foot rot, leaf rot, miscellaneous and healthy leaf of betel.

3.3 Data Collection

Data collection is the most challenging task for us. After the disease study, we have visited many places and collect images of betel leaf. We have collected images during the whole year to make a better dataset for our research work as weather change also affects the cultivation of betel leaf. During the rainy season, diseases of betel leaf increased rapidly.

3.4 Dataset Description

In our research, we have used a total of 9983 images of betel leaf, these are divided into four classes. Four classes are foot rot, leaf rot, miscellaneous, and healthy. Across all our processes, we have used color images. We have used 224 × 224 and 299 × 299 pixels images in our research work. Class-wise distribution of the collected dataset is shown in below in Table 3.1.

Table 3.1: The class-wise distribution of the collected images.

Class	Frequency
Foot rot	2101
Leaf rot	2787
Miscellaneous	683
Healthy	4412
Total	9983

Using Pillow, we have reshaped all images to 224 × 224, 299 × 299 and also decrease the quality of all images. Pillow is used for opening, manipulating, and saving many different image file formats. It is a free library of Python programming language.

3.5 Statistical Analysis

We have collected a total of 9983 images of betel leaf from different places. We have collected 5571 images of the affected leaf and 4412 images of the healthy leaf. The number of affected and healthy images of betel leaf is shown below in Fig. 3.3. Among 5571 affected images, 2101 are foot rot and 2787 are leaf rot affected images. The number of foot rot, leaf rot, miscellaneous, and healthy images of betel leaf is shown below in Fig. 3.4.

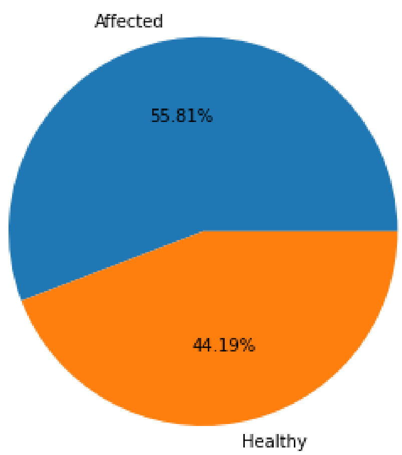

Fig. 3.3: Percentage of affected and healthy images.

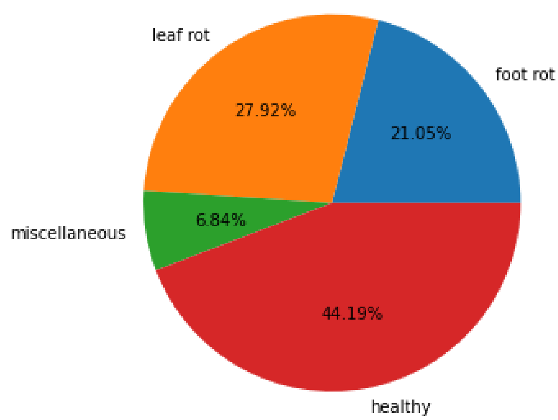

Fig. 3.4: Percentage of foot rot, leaf rot, miscellaneous, and healthy images.

3.6 CNN Model for Disease Recognition

CNN is arguably the most popular architecture of deep learning in recent times. The popularity of CNN is increasing day by day for its effectiveness and immense success. The interest in CNN started to grow after 2012 due to the introduction of AlexNet [28]. CNNs are widely used for image segmentation, image classification, and image recognition these days. The main reason behind using CNN is to process different types of data for example 1D for signal data and time series, 2D for images or audio signals, and 3D for video [29]. CNN is now used everywhere. We have used CNN in our research for its effectiveness and better performance than other algorithms. Researchers now use CNN to solve various types of problems. On the other hand, CNN models can run on any device. This also makes them universally attractive, it performs parameter sharing and uses special convolution and pooling operations. CNN can achieve higher accuracy than others. CNN can reduce image dimension during feature extraction without losing features.

Two significant parts of CNN are feature extraction and classification. A chain of convolution, pooling operations, and fully connected layers are utilized during the image classification. All processes of the feature extraction part are performed in convolution and pooling layers and classification is done in fully connected layers. Input images are passed through the convolutional layer for extracting features from the image [30]. To normalize the output of the neural network softmax is used for multiclass classification problems. CNN have three types of layers and these are input layer, hidden layers and output layer.

In subsequent sections as below, the constituent layers of CNN are explained.

3.6.1 Convolutional layer

A mathematical operation is performed in the convolutional layer which is known as convolution. It contains a set of independent filters. It is the main building block of CNN architecture. Each filter is convolved with the input image independently to obtain a feature map. The convolutions operation on an input map of 5 × 5 with a 3 × 3 filter, which creates a 3 × 3 feature map is shown below in Fig. 3.5.

1×1	1×0	1×1	0	0
0×0	1×1	1×0	1	0
0×1	0×0	1×1	1	1
0	0	1	1	0
0	1	1	0	0

Input × Filter

4		

Feature Map

Fig. 3.5: Convolution operation of 3 × 3 filter on an input map of 5 × 5.

Using the convolution of the input maps with filter and then adding a bias term, applying a nonlinear function later, an output feature map is created. The convolutions layer make CNN be a scaled variant.

The activation function helps CNN to perform complicated tasks and make CNN capable of learning. It is a remarkable part of CNN. The summed weighted transformed into input into the activation of the node by activation functions. This function is also known as the transfer function. To determine output, active functions are applied to neural networks. Activation functions chose whether the information received by the neuron should be ignored or accepted. The rectified linear unit activation (ReLU) function is the most popular activation function which is used in various kind of research work. Besides ReLU, some other common active functions are sigmoid or logistic activation function, tanh or hyperbolic tangent activation function. ReLU gives more advantage than other types of activation functions as it does not activate all the neurons at a time. CNN can easily back propagate the errors using ReLU as it is a nonlinear function. The weights are not updated and the gradient is zero for negative inputs during backpropagation, dead neurons never be activated for this. ReLU is shown below in Fig. 3.6.

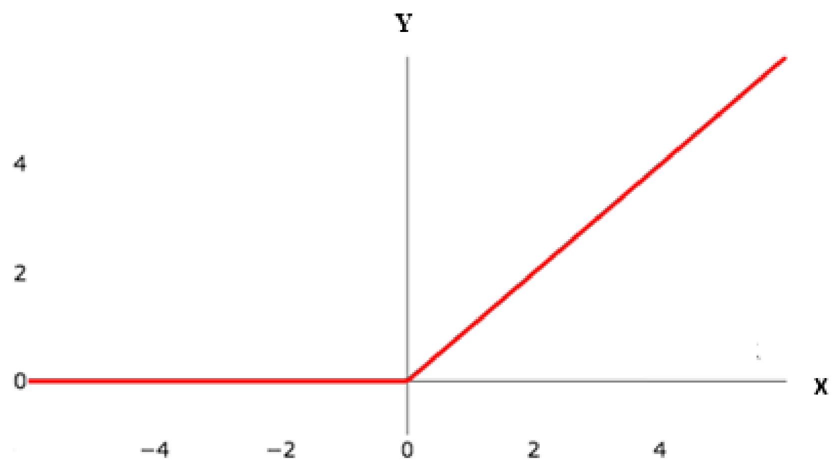

Fig. 3.6: ReLU operation.

3.6.2 Polling layers

Another building block of CNN is the pooling layer which operates on each feature map independently. In a neural network, the pooling layer reduces the spatial size of the representation to reduce the number of parameters and computations. The pooling layer is inserted between two successive convolution layers to control overfitting. Pooling operation resizes spatially of every layer of the input. These layers provide an approach for downsampling feature maps using summarization of the presence of features in patches of the feature map. Various types of spatial pooling are commonly used such as max, min, sum, average, etc. We have used three MaxPooling2D layers in our research. From the

region of the feature map, it selects the maximum element covered by the filter. From the previous feature map, the output of this layer contains the most prominent features.
A matrix of size 4 × 4 is used as initial input and a matrix size of 2 × 2 represents a filter that will run over the initial input is shown below in Fig. 3.7.

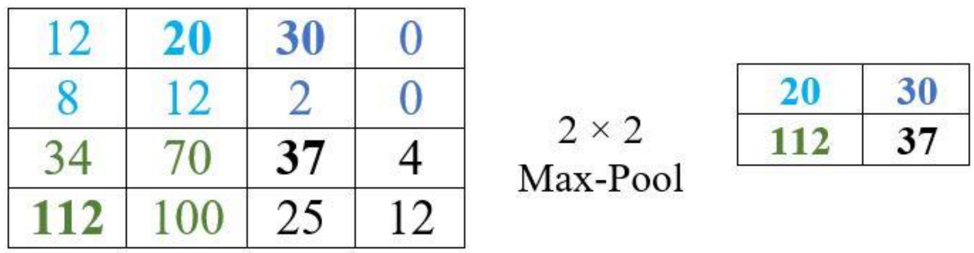

Fig. 3.7: Max polling operation with 2 × 2 filters and stride 2.

3.6.3 Fully connected layers

Fully connected layers are the last few layers in most CNN architecture. To classify the features the output of the max pooling layer is fed into the classifier in fully connected layers. The feed forward neural network is the second most time-consuming layer which is a series of fully connected layers of the neural network. Every neuron in the current layer is connected to every neuron in the previous layer in the fully connected layers. In the last fully connected layer, the number of classes to be predicted is equal to the number of neurons. A number of these layers are needed to cover the CNN architecture. From convolution layers and subsampling layers, a fully connected layer combines all the features extracted. In a feed forward neural network using high level features of the convolution layer, the image flattens into a column vector. The dropout is a simple way to prevent the neural network from overfitting. In a neural network, dropout means to drop out both hidden and visible units. Dropout has been used in the fully connected layer to prevent overfitting in our research. To reduce overfitting, it offers an effective regularization method and also improves generalization error in CNN. The final fully connected layer uses a softmax activation function, it is a more generalized logistic activation function for multiclass classification in CNNs. This function classifies the object using probability within the value of 0 and 1. In Fig. 3.8 shown below, the left network is original, and the right network is a dropped out neural network.

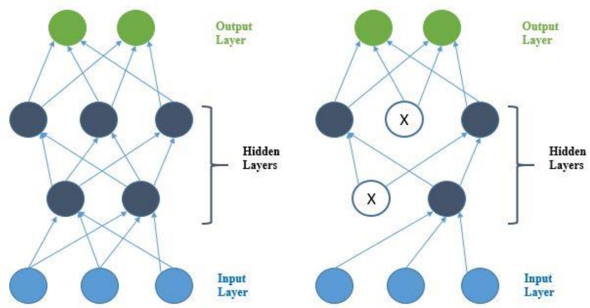

Fig. 3.8: Original and dropped out neural network.

3.6.4 Learning algorithm

In the first convolution layer curves, edges, corners, lines, and other low-level features are extracted. In the next level of convolutions layers, global features are learned. CNN learns features using back-propagation which is a training process. The four distinct sections of the back-propagation learning algorithm are forward pass, loss function, backward pass, and weight update. During the forward pass, CNN takes the training images and passes images through the whole network. Firstly randomly all the weights or filter values are initialized. In the first convolution layer, CNN is not able to investigate the low-level features. So in the classification, CNN can't take any proper decision. To update the weights of the neural network, calculation of the gradients is done by a loss function. Some commonly use loss functions are categorical crossentropy, binary crossentropy, sparse categorical crossentropy, and mean squared error. The loss will be very high for the first few training images. To make CNN able to predict the true label, the amount of loss is minimized. To find out which inputs most directly contribute to the loss of the network, a backward pass is needed to be performed through the network. In CNN over the full training set, the stochastic gradient descent is used to overcome the high cost of running backpropagation. Until the loss function reaches a threshold, the process of the forward pass, backward pass, loss function, and weight update is repeated. The network has trained well enough as the weights of layers are tuned correctly.

3.7 Implementation Requirements

To capture images of betel leaf, we have used the camera of a smartphone. After then, we have used Jupyter Notebook to resize and decrease the quality of images. Due to limitations of RAM, we have decreased the quality of images. We store our image dataset in Google Drive. We train our CNN models using Google Colab. We have used Python 3.8 in our research work.

3.8 Performance Metrics

To evaluate the recognition performance of proposed models, we have used a confusion matrix and accuracy.

Accuracy for finding average recognition performance of proposed models,

$$Accuracy = \frac{Number\ of\ correct\ predictions}{Total\ number\ of\ predictions}$$

Precision, recall, f1-score, and accuracy for finding recognition performance of each class of dataset,

$$Precision = \frac{TP}{TP + FP} \qquad Recall = \frac{TP}{TP + FN}$$

$$F1 - Score = \frac{2 \times Precision \times Recall}{Precision + Recall} \qquad Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

TP = True positive, TN = True negative, FP = False positive, FN = False negative

CHAPTER 4
Experimental Results and Discussion

4.1 Introduction

During the whole year, we have collected images of betel leaf from a different place. We have used cloud-based storage to store our dataset and model. We have also used a cloud-based notebook to train our model. In our research, we have used CNN to establish a model that will be able to recognize diseases of betel leaf. We have achieved a good accuracy using our proposed CNN model named Betel Leaf Convolutional Neural Network (BLCNN). Besides BLCNN, we have used three pre-trained models. We have used VGG16, Inception v3 and EfficientNet B0. In this research work, EfficientNet B0 has performed better than other pre-trained models. During our research, we deeply focused on our dataset to create a better model. In this chapter, we are going to describe our experiment result briefly.

4.2 Experimental Results & Analysis

Using a dataset of 9983 images with four different classes, the research is carried out. We have used Google Colab to train our models. We have used 3×3 convolutional filter in every models in this research. In the below sections, from 4.2.1 to 4.2.4, we have discussed briefly all models with related information.

4.2.1 BLCNN

Before using the pre-trained models, we have deeply focused on building a new CNN model for this research. After 200 epochs, the proposed BLCNN has achieved 90.75% training accuracy and 89.44% test accuracy. The input image size of BLCNN is

224 × 224 pixels. To control the learning rate we have used dropout and ReduceLROnPlateau callback with a lower bound of 0.00001 on the learning rate. In this model, we have used sparse categorical crossentropy as a loss function and stochastic gradient descent (sgd) as an optimizer. Using the confusion matrix module of scikit-learn, we have generated a confusion matrix of the proposed BLCNN. Confusion matrix of proposed BLCNN is shown in below Table 4.1.

Table 4.1: Confusion matrix of proposed BLCNN.

Actual Class	Predicated Class				
	Foot rot	Healthy	Leaf rot	Miscellaneous	Support
Foot rot	171	8	9	3	191
Healthy	13	371	7	9	400
Leaf rot	9	14	221	6	250
Miscellaneous	6	7	4	42	59

The confusion matrix represents the effectiveness of our proposed BLCNN without using any pre-trained model. Class-wise classification performance of BLCNN is shown in below Table 4.2.

Table 4.2: Class-wise classification performance of proposed BLCNN.

Class	Precision (%)	Recall (%)	F1-score	Accuracy (%)
Foot rot	89.52	85.92	0.87	94.67
Healthy	92.75	92.75	0.92	93.56
Leaf rot	88.40	91.70	0.90	94.56
Miscellaneous	71.18	70.00	0.71	96.11

The plot graph of the proposed CNN is shown below in Fig. 4.1.

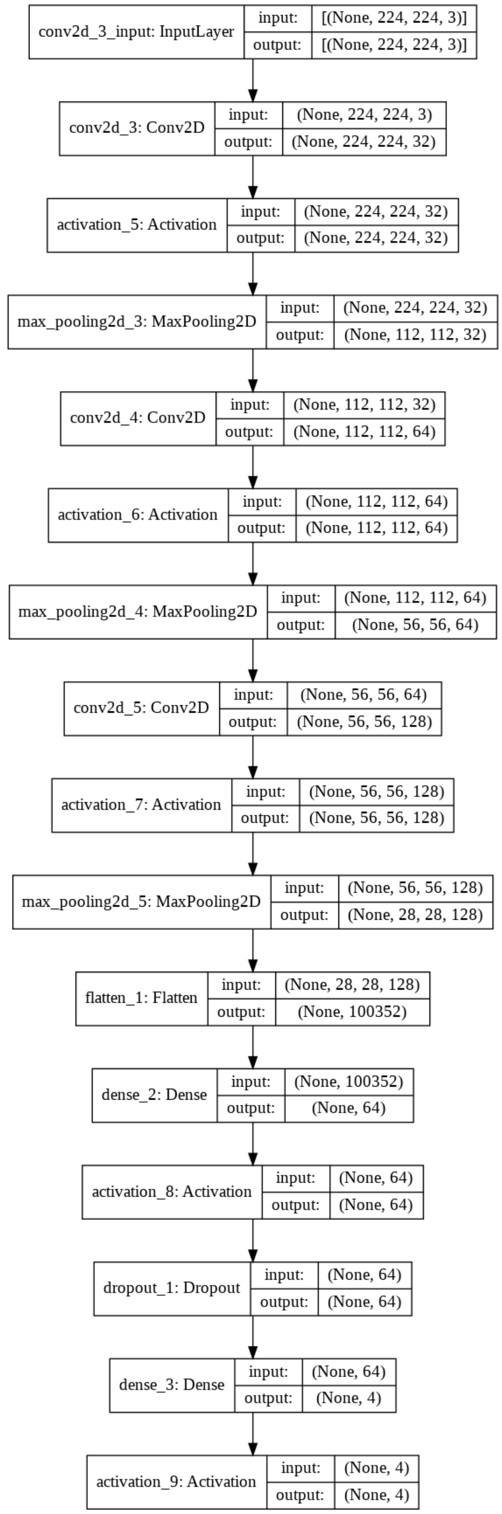

Fig. 4.1: Plot of proposed BLCNN graph.

BLCNN has used total 6,516,100 params with zero non-trainable params. We have given a model summary of our proposed BLCNN below in Fig 4.2.

```
Model: "sequential_4"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv2d_10 (Conv2D)           (None, 224, 224, 32)      896
_________________________________________________________________
activation_16 (Activation)   (None, 224, 224, 32)      0
_________________________________________________________________
max_pooling2d_10 (MaxPooling (None, 112, 112, 32)      0
_________________________________________________________________
conv2d_11 (Conv2D)           (None, 112, 112, 64)      18496
_________________________________________________________________
activation_17 (Activation)   (None, 112, 112, 64)      0
_________________________________________________________________
max_pooling2d_11 (MaxPooling (None, 56, 56, 64)        0
_________________________________________________________________
conv2d_12 (Conv2D)           (None, 56, 56, 128)       73856
_________________________________________________________________
activation_18 (Activation)   (None, 56, 56, 128)       0
_________________________________________________________________
max_pooling2d_12 (MaxPooling (None, 28, 28, 128)       0
_________________________________________________________________
flatten_4 (Flatten)          (None, 100352)            0
_________________________________________________________________
dense_7 (Dense)              (None, 64)                6422592
_________________________________________________________________
activation_19 (Activation)   (None, 64)                0
_________________________________________________________________
dropout_4 (Dropout)          (None, 64)                0
_________________________________________________________________
dense_8 (Dense)              (None, 4)                 260
_________________________________________________________________
activation_20 (Activation)   (None, 4)                 0
=================================================================
Total params: 6,516,100
Trainable params: 6,516,100
Non-trainable params: 0
_________________________________________________________________
```

Fig 4.2: Model summary of proposed BLCNN.

Accuracy and loss of both training and validation of BLCNN are shown below in Fig. 4.3 and Fig. 4.4.

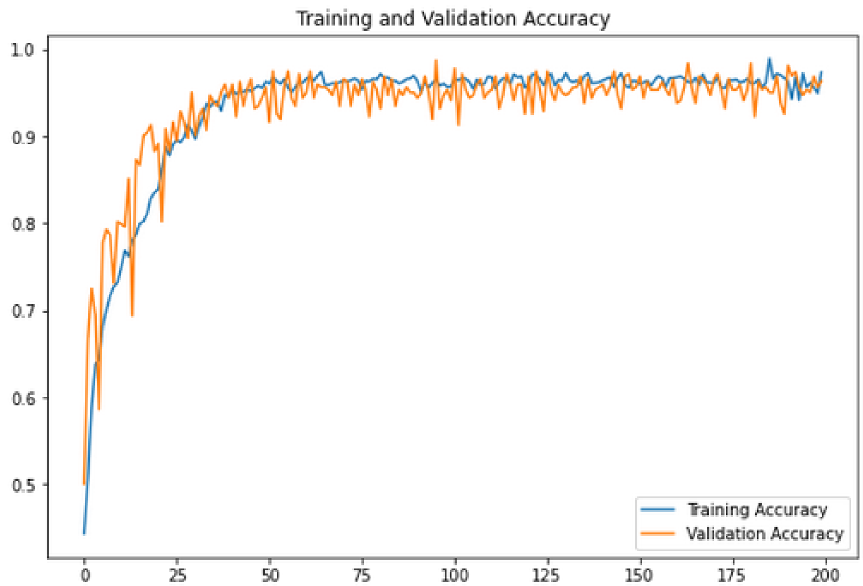

Fig. 4.3: Training accuracy vs. validation accuracy of BLCNN.

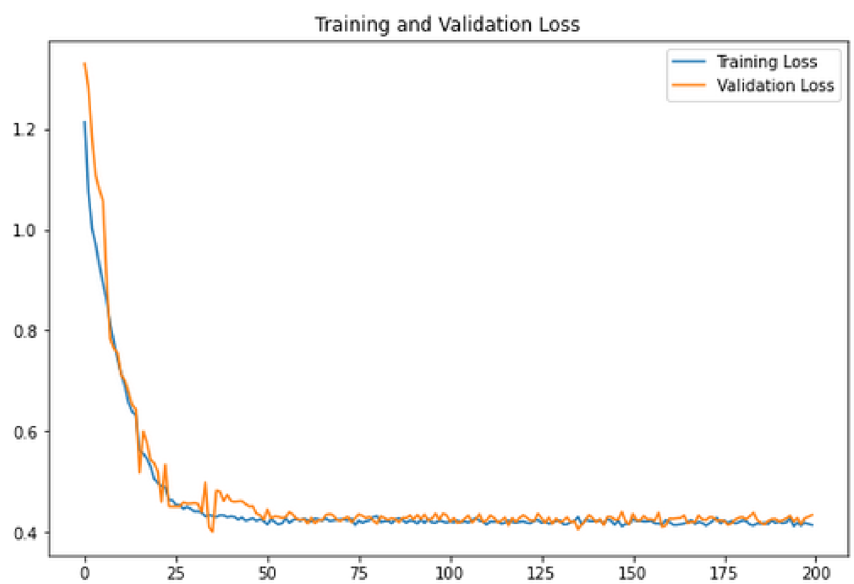

Fig. 4.4: Training loss vs. validation loss of BLCNN.

4.2.2 VGG16

VGG16 is a CNN architecture that has won ILSVRC-2014, it has 138 M (approx) parameters. Its input layer takes images with 224 × 224 pixels and it has 16 layers. This model trained with 14 million images using NVIDIA Titan Black GPU’s. VGG16 model is proposed by K. Simonyan and A. Zissermantrained. With stride 1, this architecture has used 3 × 3 filters in all convolution layers. It has used 3 fully connected layers, the first two with ReLu activation function and the last one with Softmax. VGG16 architecture map is shown below in Fig. 4.5.

Fig. 4.5: Architecture map of VGG16.

After 200 epochs using VGG16, we have achieved 93.02% training accuracy and 92.78% test accuracy. Confusion matrix of modified VGG16 is shown in below Table 4.3.

Table 4.3: Confusion matrix of proposed VGG16.

Predicated Class	Actual Class				
	Foot rot	Healthy	Leaf rot	Miscellaneous	Support
Foot rot	174	5	11	1	191
Healthy	11	383	4	2	400
Leaf rot	3	9	233	5	250
Miscellaneous	6	4	4	45	59

Modified VGG16 has also shown satisfactory performance in classifying images of all classes. Class-wise classification performance of modified VGG16 is shown in below Table 4.4.

Table 4.4: Class-wise classification performance of VGG16.

Class	Precision (%)	Recall (%)	F1-score	Accuracy (%)
Foot rot	91.10	89.69	0.90	95.88
Healthy	95.75	95.51	0.95	96.11
Leaf rot	93.20	92.46	0.93	96.00
Miscellaneous	76.27	84.91	0.80	97.56

The last layer of VGG16 contains 4097000 parameters, and the output shape is (None, 1000). To use this pre-trained model, we have removed the last layer from VGG16 and add a dense layer. Output of the last layer of this modified VGG16 is (None, 4), this layer contains 16388 parameters. Last three fully connected layers of modified VGG16 is shown in below Fig. 4.6.

```
fc1 (Dense)              (None, 4096)          102764544

fc2 (Dense)              (None, 4096)          16781312

dense_24 (Dense)         (None, 4)             16388
```

Fig. 4.6: Last three fully connected layers of modified VGG16.

Accuracy and loss of both training and validation of modified VGG16 are shown below in Fig. 4.7 and Fig. 4.8.

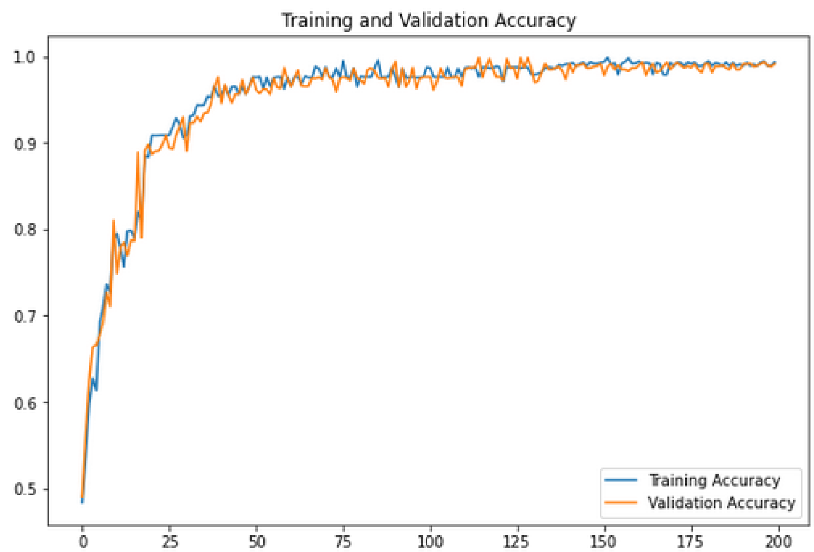

Fig. 4.7: Training accuracy vs. validation accuracy of modified VGG16.

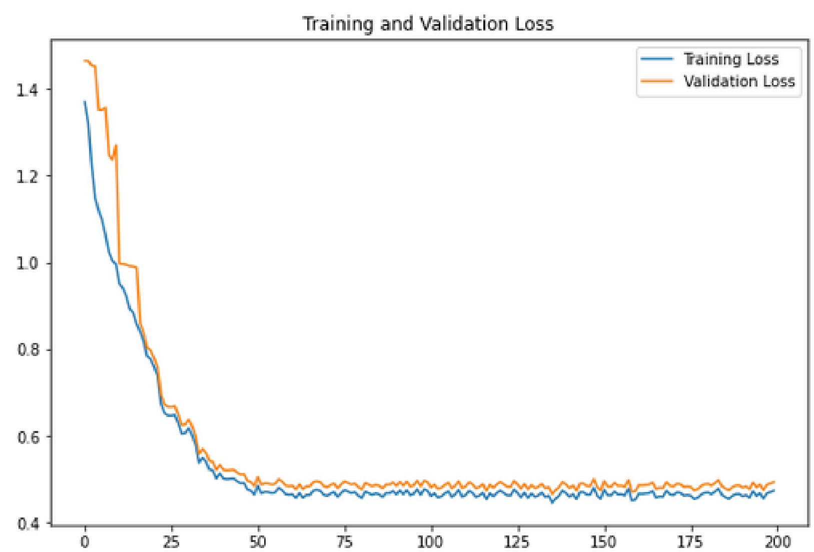

Fig. 4.8: Training loss vs. validation loss of modified VGG16.

4.2.3 Inception V3

Inception V3 is a CNN architecture, it is widely used in the field of image recognition for its efficiency. Before work with Inception V3, we have resized all images to 299 × 299 pixels. Inception V3 contains a total of 42 layers and the output of the last layer is (None, 8, 8, 2048). It was developed by Google and the idea of factorization is introduced here. The softmax activation function is used in its last layer. It can reduce the number of connections and parameters of the network without reducing efficiency. Schematic representation of modified Inception V3 is shown in below Fig. 4.9.

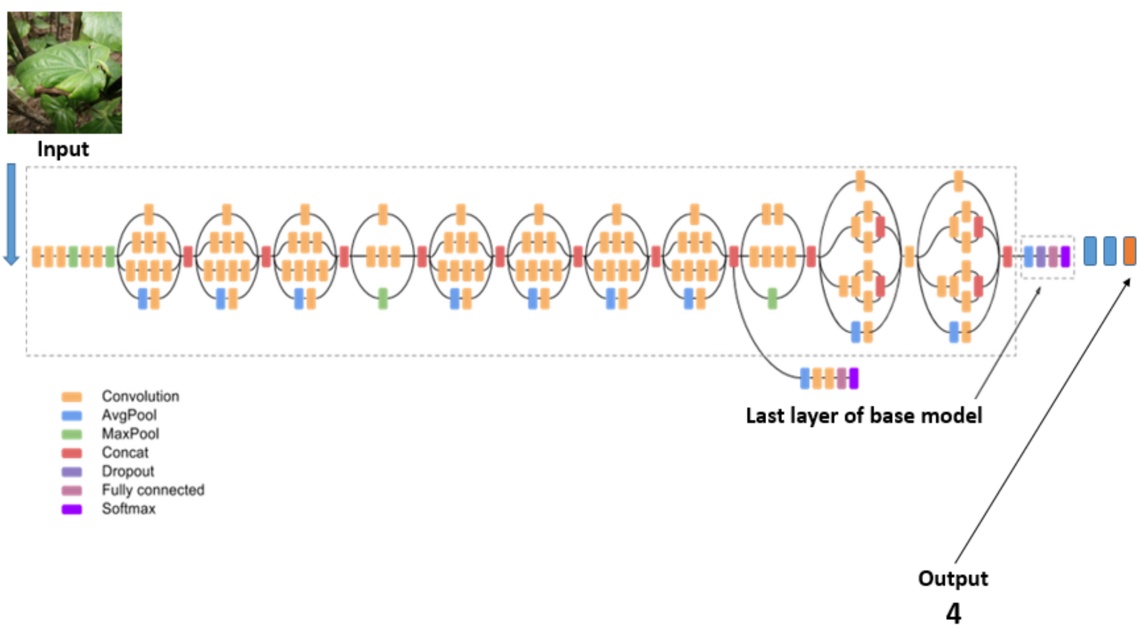

Fig 4.9: Schematic representation of modified Inception V3.

We have modified this Inception V3 architecture. We have added this model as a base model in one layer. After this layer, we have added three new layers. The last layer of this modified architecture is a dense layer and its output shape is (None, 4). Using this modified pre-trained model, we have achieved 95.18% training accuracy and 94.55% test accuracy with 200 epochs. Training accuracy is increased by 2.16% and test accuracy is increased by 1.77% after using Inception V3. Confusion matrix of modified Inception V3 is shown in below Table 4.5.

Table 4.5: Confusion matrix of modified Inception V3.

Predicated Class	Actual Class				
	Foot rot	Healthy	Leaf rot	Miscellaneous	Support
Foot rot	178	2	7	4	191
Healthy	8	387	5	0	400
Leaf rot	5	4	238	3	250
Miscellaneous	2	6	3	48	59

Modified Inception has shown better performance in classifying images of all classes than modified VGG16. Class-wise classification performance of modified Inception V3 is shown in below Table 4.6.

Table 4.6: Class-wise classification performance of modified Inception V3.

Class	Precision (%)	Recall (%)	F1-score	Accuracy (%)
Foot rot	93.19	92.23	0.93	96.88
Healthy	96.75	96.99	0.97	97.22
Leaf rot	95.20	94.07	0.95	97.00
Miscellaneous	81.36	87.27	0.84	98.00

The last layer of modified Inception V3 contains 8196 parameters. Before this dense layer, one global average pooling layer, and one dropout has added. Before these three layers, we have added the Inception V3 model as a layer, this layer contains 21802784 parameters.

Model summary of modified Inception V3 is shown below in Fig. 4.10.

```
Model: "sequential_2"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
inception_v3 (Model)         (None, 8, 8, 2048)        21802784
_________________________________________________________________
global_average_pooling2d_2 ( (None, 2048)              0
_________________________________________________________________
dropout_2 (Dropout)          (None, 2048)              0
_________________________________________________________________
dense_1 (Dense)              (None, 4)                 8196
=================================================================
Total params: 21,810,980
Trainable params: 8,196
Non-trainable params: 21,802,784
_________________________________________________________________
```

Fig 4.10: Summary of modified Inception V3.

Accuracy and loss of both training and validation of modified Inception V3 are shown below in Fig. 4.11 and Fig. 4.12.

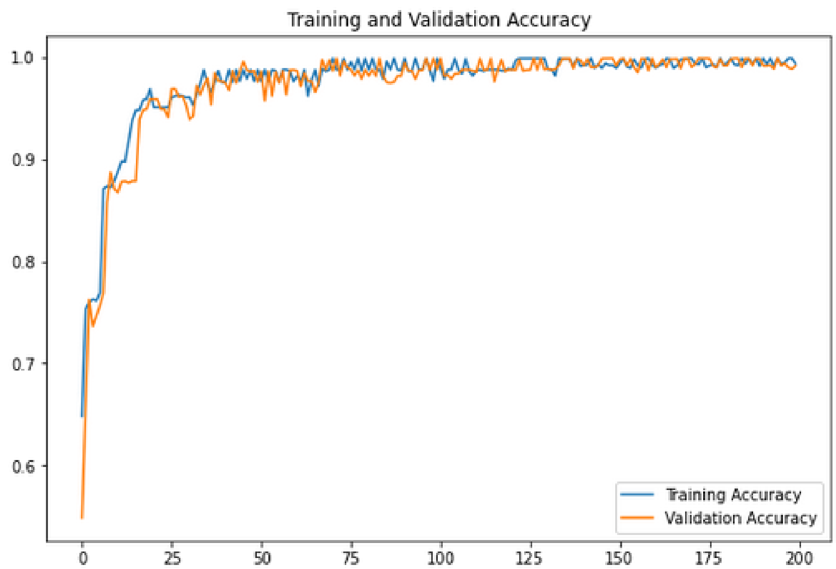

Fig. 4.11: Training accuracy vs. validation accuracy of modified Inception V3.

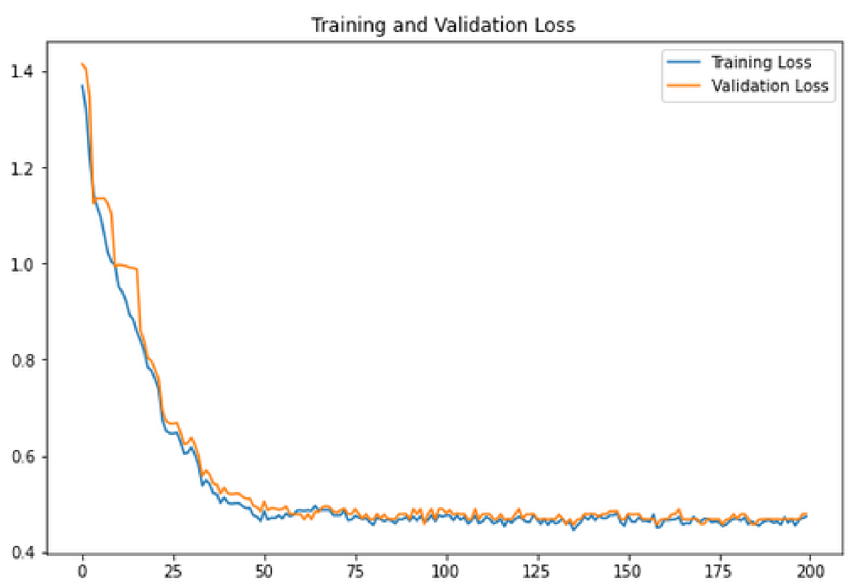

Fig. 4.12: Training loss vs. validation loss of modified Inception V3.

4.2.4 EfficientNet B0

EfficientNet is a group of CNN models, this group contains a total of 8 models. These are B0, B1, B2, B3, B4, B5, B6, and B7. In this research, we have used the B0 model of EfficientNet. This has achieved 84.4% accuracy in the ImageNet classification problem with 66 M parameters. Instead of ReLU, EfficientNet uses a new activation function known as Swish. Based on inverted bottleneck residual blocks of MobileNetV2, the base of the EfficientNet B0 network is developed. The modified architecture of EfficientNet B0 is shown is below in Fig. 4.13.

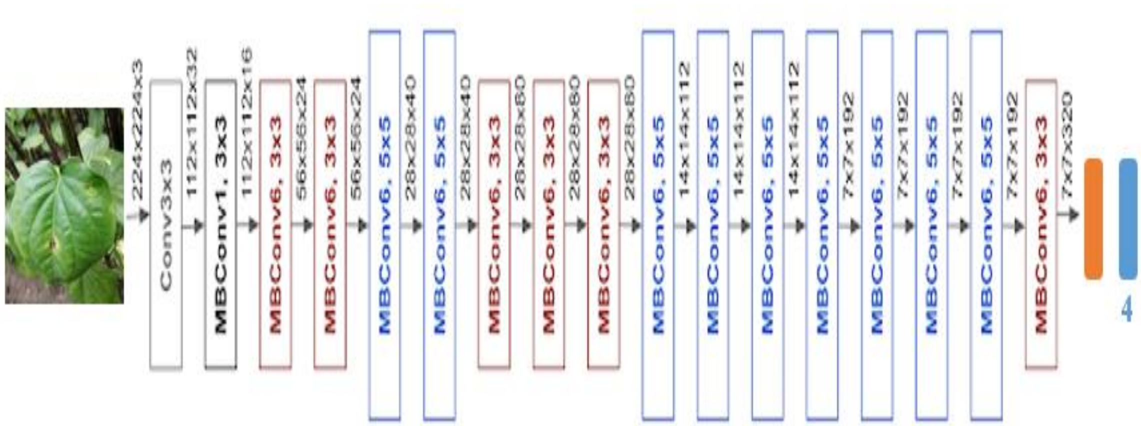

Fig. 4.13: The modified architecture of EfficientNet B0 used in this work.

The output of the last layer of EfficientNet B0 is (None, 1000) with 1281000. We have added two more with this pre-trained model. We have added a dropout layer and a dense layer. After this, the output of the last layer is (None, 4) with 4004 params. We have used 224 × 224 pixels images as the input image. This modified EfficientNet B0 architecture has shown satisfactory performance in this research. In every class, it has achieved a remarkable accuracy than BLCNN and other pre-trained models. The previous 3 models haven't performed well in recognizing miscellaneous class images. But this model has achieved good accuracy in this class. We have achieved 97.24% training accuracy and

96.77% test accuracy after 200 epochs. The training accuracy is increased by 2.06% and test accuracy is increased by 2.22% using this model B0 model of EfficientNet. The confusion matrix of this model represents its efficiency in recognizing images.

Confusion matrix of modified EfficientNet B0 architecture is shown in below Table 4.7.

Table 4.7: Confusion matrix of modified EfficientNet B0 architecture.

Predicated Class	Actual Class				
	Foot rot	Healthy	Leaf rot	Miscellaneous	Support
Foot rot	184	2	4	1	191
Healthy	5	392	2	1	400
Leaf rot	3	2	243	2	250
Miscellaneous	4	2	1	52	59

Modified EfficientNet B0 has shown well performance in classifying images of all classes than other models. Class-wise classification performance of modified EfficientNet B0 is shown in below Table 4.8.

Table 4.8: Class-wise classification performance of modified EfficientNet B0.

Class	Precision (%)	Recall (%)	F1-score	Accuracy (%)
Foot rot	96.34	93.88	0.95	97.89
Healthy	98.00	98.49	0.98	98.44
Leaf rot	97.20	97.20	0.97	98.45
Miscellaneous	88.14	92.86	0.90	98.78

This modified EfficientNet B0 has given the best performance than BLCNN and the other two pre-trained models. The performance of BLCNN is also satisfying us in recognizing unknown images. The inverted bottleneck MBConv is the main building of EfficientNet. It is first conducted in MobileNetV2. After modifying EfficientNet B0, the number of the total parameters is 5,334,575. Among these 4,004 are trainable parameters and 5,330,571 are non-trainable parameters.

Model summary of modified EfficientNet B0 is shown below in Fig. 4.14.

```
Model: "sequential_2"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
efficientnetb0 (Functional)  (None, 1000)              5330571
_________________________________________________________________
dropout (Dropout)            (None, 1000)              0
_________________________________________________________________
dense (Dense)                (None, 4)                 4004
=================================================================
Total params: 5,334,575
Trainable params: 4,004
Non-trainable params: 5,330,571
_________________________________________________________________
```

Fig. 4.14: Model summary of modified EfficientNet B0 architecture.

Accuracy and loss of both training and validation of modified EfficientNet B0 are shown below in Fig. 4.15 and Fig. 4.16.

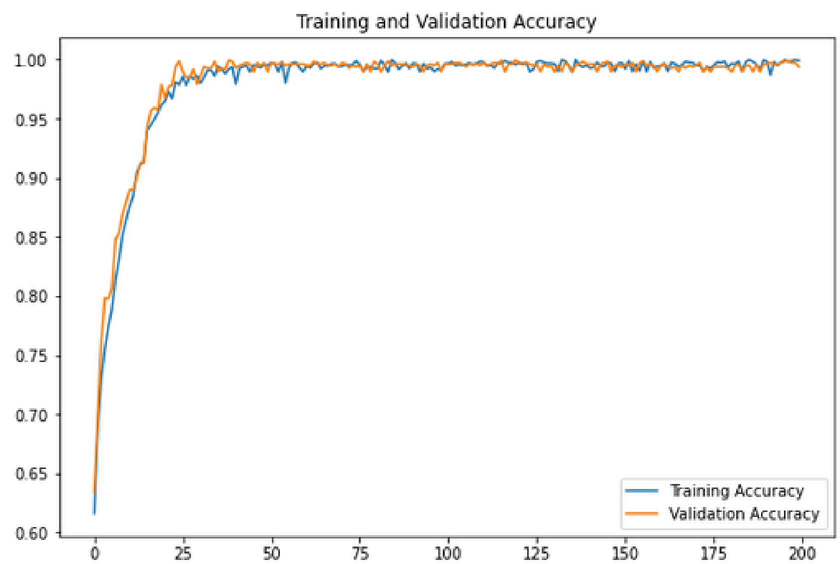

Fig. 4.15: Training accuracy vs. validation accuracy of modified EfficientNet B0.

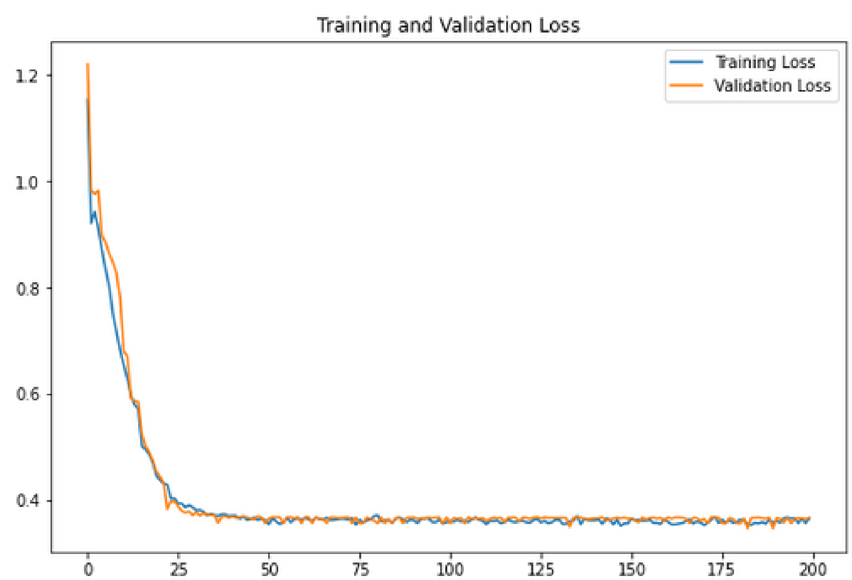

Fig. 4.16: Training loss vs. validation loss of modified EfficientNet B0.

4.3 Comparative Analysis

We have used two different size input images in this research. We have given input image size with total numbers of parameters of four models below in Table 4.9.

Table 4.9: The default input image sizes and number of total parameters of four models.

Model name	Image size	Number of total parameters
BLCNN	224 × 224	6,516,100
VGG16	224 × 224	134,276,932
Inception V3	299 × 299	21,810,980
EfficientNet B0	224 × 224	5,334,575

EfficientNet B0 has shown the best performance than other models. The training and test accuracy of different models is shown below in Table 4.10.

Table 4.10: The performance of four different models.

Model name	Number of epochs	Training accuracy (%)	Test accuracy (%)
BLCNN	200	90.75	89.44
VGG16	200	93.02	92.78
Inception V3	200	95.18	94.55
EfficientNet B0	200	97.24	96.77

During this research, we have found that all models have done well in recognizing images of all class.

Class-wise performance of different models is shown in below Table 4.11.

Table 4.11: Class-wise performance of four models.

Class name	Model name			
	BLCNN (%)	VGG16 (%)	Inception V3 (%)	EfficientNet B0 (%)
Foot rot	94.67	95.88	96.88	97.89
Healthy	93.56	96.11	97.22	98.44
Leaf rot	94.56	96.00	97.00	98.45
Miscellaneous	96.11	97.56	98.00	98.78

We have compared our research work with others also. Table 4.12 shows the comparative performances of all methods related to our work. We have used a large image dataset, which helps us to achieve high accuracy.

Table 4.12: Comparative performances among different works on betel leaves.

Work Done	Mode of Solution	Size of Dataset	Segmentation Algorithm	Classification Status	Classifier	Accuracy
BLCNN (This work)	Deep learning	9983	CNN	√	CNN	90.75
VGG16 (This work)	Deep learning	9983	CNN	√	CNN	93.02
Inception V3 (This work)	Deep learning	9983	CNN	√	CNN	95.18

EfficientNet B0 (This work)	Deep learning	9983	CNN	√	CNN	97.24
Tamilsankar et al. [8]	Image processing	100	Watershed transformation algorithm	√	Minimum Distance Classifier	*NM*[1]
Jayanthi et al. [9]	Image processing	*NM*	L*a*b* color space model, watershed transformation algorithm	√	Multiclass SVM	95.85%
Dey et al. [10]	Image processing and traditional machine learning	12	Otsu thresholding	×	*NA*[1]	*NA*
Ramesh et al. [12]	Image processing	*NA*	*K*-means clustering	√	Fuzzy classifier	*NM*

[1]*NM*: Not Mentioned.

[2]*NA*: Not Applicable.

We have visualized the performance of four models with every class using two bar plots below, in Fig. 4.17 and Fig. 4.18.

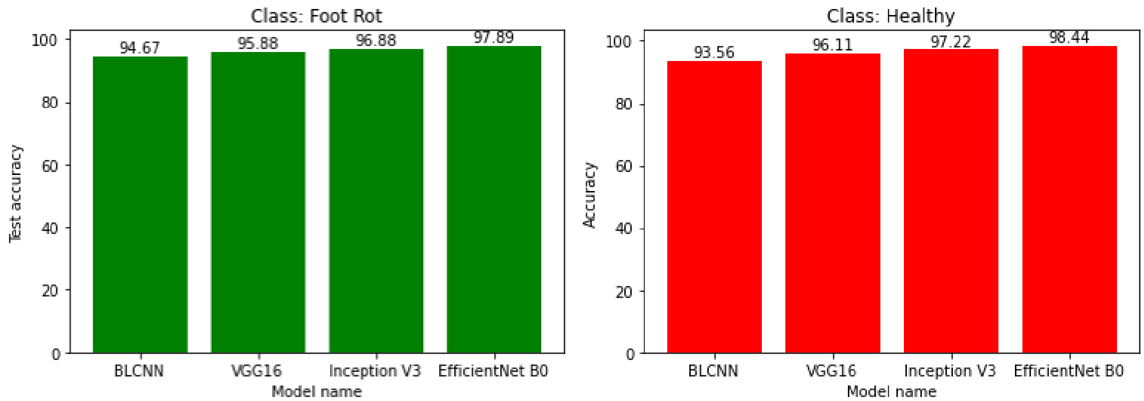

Fig. 4.17: Performance of four models in foot rot and healthy class.

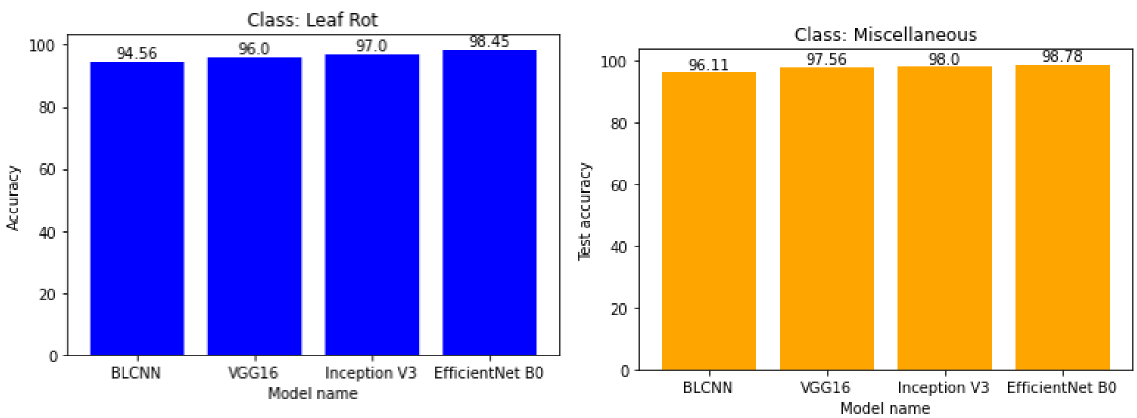

Fig. 4.18: Performance of four models in leaf rot and miscellaneous class.

In below, Fig. 4.19 shows the training and test accuracy of each model.

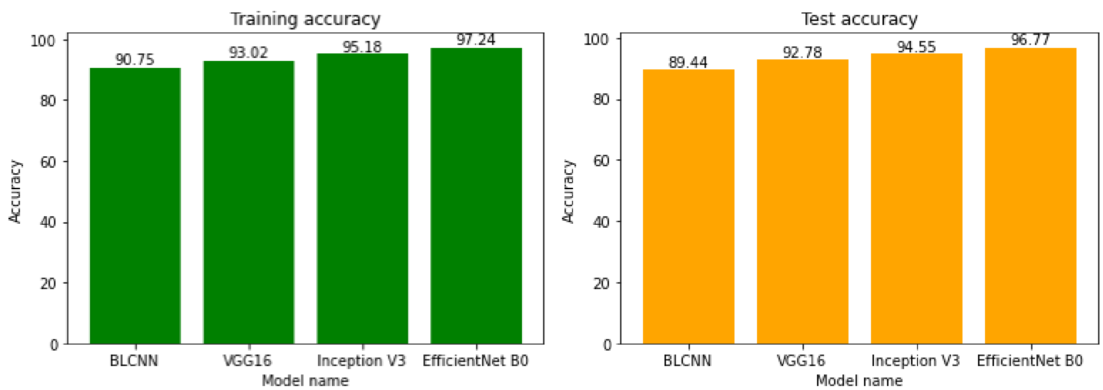

Fig. 4.19: The training and test accuracy of each model.

4.4 Discussion

In this section, we explain the performance of our proposed CNN models using the accuracy and confusion matrix. The confusion matrix shows the effectiveness of our proposed CNN models to recognize diseases separately. Using the model summary and different graphs of four models, we have explained the proposed BLCNN and the other three pre-trained models. Proposed BLCNN has achieved 90.75% training accuracy and 89.44% test accuracy. But EfficientNet B0 has performed better in recognizing images of all classes. The training accuracy of modified EfficientNet B0 is 97.24% and test accuracy is 96.77%. The other two pre-trained models also have performed well and achieved high accuracy.

CHAPTER 5

Impact on Society, Environment, and Sustainability

5.1 Impact on Society

One of the main goal of this research is to enhance financial condition of people who involved with betel leaf cultivation. Farmers of Bangladesh, as well as other countries of South Asia, are unfamiliar with modern technologies of agriculture. For this reason, they face financial loss most of the time. Plants of betel are very sensitive, so the cultivation of betel leaf is not an easy task. To earn a good profit, farmers need to depend on their luck. The disease of betel leaf makes farmer life miserable. Our research will help farmers to recognize betel leaf diseases at an early stage. Then they will be able to take action to control disease and ensure their profit. In this way, our research will help farmers to financially benefit. This disease recognition system will also help to ensure quality production. We have trained the CNN models with a better image dataset and achieved high accuracy. CNN is compatible with modern technologies, the model can be easily deployed on any platform. So, farmers or ordinary people who have not enough knowledge about betel leaf diseases will be benefited from our research. So, this research has a great impact on society.

5.2 Impact on Environment

To control the diseases of betel leaf farmers use different types of fertilizers. Fertilizers are a great threat to the environment. These fertilizers pollution water, air, and solid of nearby lands. Large numbers of farmers are earned their livelihood from betel leaf cultivation. As a result, the use of fertilizers in this cultivation is growing day by day. But fertilizers can't fully control the diseases. This increases the cost of cultivation. To solve this problem, farmers need to recognize diseases at an early stage and then take proper steps to control diseases. This will decrease the use of fertilizer and increase the profit of betel leaf cultivation. So, our research has a great impact on the environment also. Our proposed CNN models can recognize diseases of betel leaf correctly. These CNN models can be used in any type of smart software. Using smart software based on this model people will be able to recognize diseases at an early stage. This will decrease the use of fertilizer and save our society from being polluted.

5.3 Sustainability Plan

The mission of our research is to help farmers to recognize betel leaf diseases at an early stage. Ministry of Agriculture and other organizations involved with agriculture can use this model to speed up their work. From our research farmers will be benefited financially, it will also impact the national economy as demands for betel leaf is growing rapidly in both local and foreign markets.

CHAPTER 6

Summary, Conclusion, Recommendation, And Implication for Future Research

6.1 Summary of the Study

In this research work, we have used deep learning to recognize betel leaf diseases. This work is divided into several parts such as Disease study, Image collection, Implementation of methodology, and Evaluation of experiment. We have collected images from different places during the whole year and then made an image dataset for our research. Using Jupyter Notebook, we have resized all images and also increase the quality of images. After reviewing different related research work, we have decided to use CNN in our research. CNN is compatible with modern technologies which is one of the main of using CNN. We train our CNN models using Google Colab. Our proposed BLCNN has achieved a good accuracy without using pre-trained model. To improve the ability of recognition, we have also used three pre-trained models. The EfficientNet B0 model has performed better than other models. After comparing with related research, we found that our CNN models is more efficient than others. These CNN models has shown a satisfactory performance with new images.

6.2 Limitations and Conclusions

In this research, we establish an efficient CNN model to recognize betel leaf diseases using EfficientNet B0. This model can recognize betel leaf diseases correctly, but we have some limitations also. After visiting many places we found two common diseases only. We have used only two diseases of betel leaf in our research work. Without two diseases, this model will not be able to recognize other diseases. In this research, we proposed a new approach for recognizing diseases of betel leaf using CNN. CNN is very popular in various fields of research. We have evaluated the performance of our CNN models, which is very satisfactory. This CNN models works very well with new test images. This model also shows its efficiency in recognizing each class separately. We deeply focused on our dataset and created a better dataset for research. Our research will help farmers who involved with betel leaf cultivation. It will enhance their financial condition by ensuring the quality of betel leaf. In local and foreign markets, demands for betel leaf is increasing incredibly for various use of this leaf. This research has a great impact on society and the environment. Using this research Ministry of Agriculture and other organizations can change the economic condition of farmers and ensure quality production of betel leaf.

6.3 Implication for Further Study

The use of AI-based systems makes our daily life easier than before. AI transforms everything like electricity. We want to deploy our CNN models in an android application. Users of the smartphone and internet are increasing day by day. A user-friendly mobile application with attractive GUIs will be very helpful for ordinary people and farmers. Now our CNN models can recognize images of two diseases. In the future, we will increase the number of diseases. A more robust dataset will be created in future studies. We will also focus on the accuracy of our BLCNN model in the future. We want to publish a model as a pre-trained model for other leaf diseases. With the help of the Ministry of Agriculture, the model can be made larger and taken forward.

REFERENCES

[1] Bangladesh houses 16.5 million farmer families, available at <<https://www.dhakatribune.com/bangladesh/agriculture/2019/10/27/agricultural-census-report-16-5-million-farmer-families-in-bangladesh>>, last accessed on 12-07-2020 at 11:00 PM.

[2] Economy of Bangladesh, available at <<https://en.wikipedia.org/wiki/Economy_of_Bangladesh>>, last accessed on 12-07-2020 at 12:00 PM.

[3] Bangladesh Literacy Rate 1981-2020, <<https://www.macrotrends.net/countries/BGD/bangladesh/literacy-rate#:~:text=Adult%20literacy%20rate%20is%20the,a%200.13%25%20increase%20from%202016>>, last accessed on 12-07-2020 at 12:30 PM..

[4] Kawaski R, Uga H, Kagiwada S, Iyatomi H, "Basic study of viral plant diseases using convolutional neural networks" In: Proceedings of the international symposium on visual computing, pp. 638–645, December 2015.

[5] Yang L, Yi S, Zebg N, Liu Y, Zhang Y, "Identification of rice diseases using deep convolutional neural networks" Neurocomputing, vol. 267, pp. 378–384, December 2017.

[6] Brahimi M, Boukhalfa K, Moussaoui A, "Deep learning of tomato diseases: classification and symptoms visualization" Appl Artif Intell, vol. 31, pp. 299-315, May 2017.

[7] P. Guha, "Betel Leaf: The Neglected Green Gold of India" J. Hum. Ecol., vol. 19, pp. 87-93, October 2017.

[8] P.Tamilsankar1, Dr.T. Gunasekar, "COMPUTER AIDED DISEASES IDENTIFICATION FOR BETEL LEAF" IRJET, vol. 2, pp. 2577-2581, December 2015.

[9] Dr. S. K. Jayanthi1, C.Lalitha, "Betel Leaf Disease Detection Using Histogram of Oriented Gradients and Multiclass SVM" IJIRCCE, vol. 5, pp. 13994-14001, August 2017.

[10] Amar Kumar Deya, Manisha Sharmaa, M.R.Meshramb, "Image Processing Based Leaf Rot Disease Detection of Betel Vine (Piper BetleL.)" Procedia Computer Science, vol. 85, pp. 748-754, February 2016.

[11] Mr.J.Vijayakumar, Dr.S.Arumugam, "Recognition of Powdery Mildew Disease for Betelvine Plants Using Digital Image Processing" IJDPS, vol.3, pp. 231-241, March 2012.

[12] L. Ramesh, V. Nirmala, “Betel Leaf Disease Classification Using Clustering Method” JCHPS, pp.193-196, February 2017.

[13] Srdjan Sladojevic, Marko Arsenovic, Andras Anderla, Dubravko Culibrk, and Darko Stefanovic, “Deep Neural Networks Based Recognition of Plant Diseases by Leaf Image Classification” Computational Intelligence and Neuroscience, vol. 2016, June 2016.

[14] Ishak, Syafiqah & Fazalul Rahiman, Mohd Hafiz & Mohd Kanafiah, Siti & Saad, Hashim, “Leaf disease classification using artificial neural network” Jurnal Teknologi (Sciences & Engineering), vol. 77, pp. 109–114, October 2015.

[15] Sabrol H., Kumar S., “Plant Leaf Disease Detection Using Adaptive Neuro-Fuzzy Classification” Advances in Intelligent Systems and Computing, vol. 943, pp. 434-443, April 2019.

[16] Yadav R., Kumar Rana Y., Nagpal S., “Plant Leaf Disease Detection and Classification Using Particle Swarm Optimization” Machine Learning for Networking. MLN 2018. Lecture Notes in Computer Science, vol. 11407, pp. 294-306, May 2019.

[17] Khan, M.A., Akram, T., Sharif, M., Tanzila S., “An automated system for cucumber leaf diseased spot detection and classification using improved saliency method and deep features selection” Multimed Tools App, vol. 79, pp. 18627–18656, February 2020.

[18] Ahila Priyadharshini, R., Arivazhagan, S., Arun, M. et al, “Maize leaf disease classification using deep convolutional neural networks” Neural Comput & Applic, vol. 31, pp. 8887–8895, May 2019.

[19] Basavaiah, J., Arlene Anthony, “A. Tomato Leaf Disease Classification using Multiple Feature Extraction Techniques” Wireless Pers Commun, vol. 115, pp. 633–651, June 2020.

[20] Goswami M., Maheshwari S., Poonia A., Songara D., “Taxonomy of Leaf Disease Detection and Classification” Recent Findings in Intelligent Computing Techniques. Advances in Intelligent Systems and Computing”, vol. 708, pp. 557-563, November 2018.

[21] Ji, M., Zhang, K., Wu, Q. et al., “Multi-label learning for crop leaf diseases recognition and severity estimation based on convolutional neural networks” Soft Comput., vol. 24, pp. 15327–15340, March 2020.

[22] Khamparia, A., Saini, G., Gupta, D. et al., “Seasonal Crops Disease Prediction and Classification Using Deep Convolutional Encoder Network” Circuits Syst Signal Process, vol. 39, pp. 818–836, January 2019.

[23] Nidhis A.D., Pardhu C.N.V., Reddy K.C., Deepa K., “Cluster Based Paddy Leaf Disease Detection, Classification and Diagnosis in Crop Health Monitoring Unit.” Computer Aided Intervention and Diagnostics in Clinical and Medical Images. Lecture Notes in Computational Vision and Biomechanics, vol. 31, pp. 281-291, January 2019.

[24] Zhang Chuanlei, Zhang Shanwen, Yang Jucheng, Shi Yancui, Chen Jia, “Apple leaf disease identification using genetic algorithm and correlation based feature selection method” Int J Agric & Biol Eng, vol. 10, pp. 74-83, March 2017.

[25] Ramakrishnan M. and Sahaya Anselin Nisha A., “Groundnut leaf disease detection and classification by using back probagation algorithm” International Conference on Communications and Signal Processing (ICCSP) 2015, pp. 0964-0968, April 2015.

[26] E. K. Ratnasari, M. Mentari, R. K. Dewi and R. V. H. Ginardi, “Sugarcane leaf disease detection and severity estimation based on segmented spots image” Proceedings of International Conference on Information, Communication Technology and System (ICTS) 2014, pp. 93-98, January 2016.

[27] N. Agrawal, J. Singhai and D. K. Agarwal, “Grape leaf disease detection and classification using multi-class support vector machine” International Conference on Recent Innovations in Signal processing and Embedded Systems (RISE), Bhopal, 2017, pp. 238-244, June 2018.

[28] AlexNet, available at <<https://en.wikipedia.org/wiki/AlexNet>>, last accessed on 12-09-2020 at 10:00 PM.

[29] Yann LeCun, Yoshua Bengio, Geoffrey Hinton, “Deep Learning” Nature, vol. 521, pp. 436-444, May 2015.

[30] Convolutional Neural Networks: The Biologically-Inspired Model, available at <<https://www.codementor.io/@james_aka_yale/convolutional-neural-networks-the-biologically-inspired-model-iq6s48zms>>, >>, last accessed on 12-09-2020 at 11:00 PM.

APPENDICES

Abbreviation

K-NN = *K*-nearest neighbors

GA = Genetic Algorithm

CFS = Correlation based Feature Selection

M = Million

Approx = approximate

Appendix: Research Reflections

At the very beginning, I have little idea about machine learning, deep learning, and recognition techniques. My supervisor is very kind and sincere. He helped me a lot and gave valuable guidance from the very beginning. During the whole time of research, we learned many things such as the procedure of making a better dataset, how to design a CNN models, and how to overcome overfitting. Finally, by doing this research, I have gained knowledge about different algorithms of machine learning and deep learning, this research also inspired me to do more in the future.

PLAGIARISM REPORT

Plagiarism Report

ORIGINALITY REPORT

13%	6%	10%	4%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	link.springer.com Internet Source	2%
2	Md. Zahid Hasan, Md. Sazzadur Ahamed, Aniruddha Rakshit, K. M. Zubair Hasan. "Recognition of Jute Diseases by Leaf Image Classification using Convolutional Neural Network", 2019 10th International Conference on Computing, Communication and Networking Technologies (ICCCNT), 2019 Publication	2%
3	Ümit Atila, Murat Uçar, Kemal Akyol, Emine Uçar. "Plant leaf disease classification using EfficientNet deep learning model", Ecological Informatics, 2021 Publication	1%
4	dspace.daffodilvarsity.edu.bd:8080 Internet Source	1%
5	"Recent Trends in Image Processing and Pattern Recognition", Springer Science and Business Media LLC, 2019	1%

Publication

6	Miaomiao Ji, Keke Zhang, Qiufeng Wu, Zhao Deng. "Multi-label learning for crop leaf diseases recognition and severity estimation based on convolutional neural networks", Soft Computing, 2020 Publication	<1 %
7	hdl.handle.net Internet Source	<1 %
8	Md. Tarek Habib, Md. Jueal Mia, Mohammad Shorif Uddin, Farruk Ahmed. "An in-depth exploration of automated jackfruit disease recognition", Journal of King Saud University - Computer and Information Sciences, 2020 Publication	<1 %
9	"Advances in Computer Vision", Springer Science and Business Media LLC, 2020 Publication	<1 %
10	"Advances in Neural Networks - ISNN 2017", Springer Science and Business Media LLC, 2017 Publication	<1 %
11	ethesis.nitrkl.ac.in Internet Source	<1 %
12	Abdala Nour, Boubakeur Boufama. "Convolutional Neural Network Strategy for Skin	<1 %

Cancer Lesions Classifications and Detections", Proceedings of the 11th ACM International Conference on Bioinformatics, Computational Biology and Health Informatics, 2020
Publication

13	Ratnasari, Evy Kamilah, Mustika Mentari, Ratih Kartika Dewi, and R. V. Hari Ginardi. "Sugarcane leaf disease detection and severity estimation based on segmented spots image", Proceedings of International Conference on Information Communication Technology and System (ICTS) 2014, 2014. Publication	<1%
14	Muhammad Attique Khan, Tallha Akram, Muhammad Sharif, Kashif Javed, Mudassar Raza, Tanzila Saba. "An automated system for cucumber leaf diseased spot detection and classification using improved saliency method and deep features selection", Multimedia Tools and Applications, 2020 Publication	<1%
15	en.wikipedia.org Internet Source	<1%
16	Submitted to University of Melbourne Student Paper	<1%
17	harvest.usask.ca Internet Source	<1%

18	Maksim Bazulin, Denis Sabitov, Marwan Charara. "VTI parameters determination from synthetic sonic logging data using a convolutional neural network", SEG Technical Program Expanded Abstracts 2020, 2020 Publication	<1 %
19	niphm.gov.in Internet Source	<1 %
20	"Smart Trends in Information Technology and Computer Communications", Springer Science and Business Media LLC, 2016 Publication	<1 %
21	Submitted to Universiti Tenaga Nasional Student Paper	<1 %
22	Rezwan Matin, Damian Valles. "A Speech Emotion Recognition Solution-based on Support Vector Machine for Children with Autism Spectrum Disorder to Help Identify Human Emotions", 2020 Intermountain Engineering, Technology and Computing (IETC), 2020 Publication	<1 %
23	Submitted to TechKnowledge Student Paper	<1 %
24	Submitted to Liverpool John Moores University Student Paper	<1 %
	Submitted to Coventry University	

25	Student Paper	<1%
26	Ramakrishnan M., Sahaya Anselin Nisha A.. "Groundnut leaf disease detection and classification by using back probagation algorithm", 2015 International Conference on Communications and Signal Processing (ICCSP), 2015 Publication	<1%
27	zombiedoc.com Internet Source	<1%
28	Ramar Ahila Priyadharshini, Selvaraj Arivazhagan, Madakannu Arun, Annamalai Mirnalini. "Maize leaf disease classification using deep convolutional neural networks", Neural Computing and Applications, 2019 Publication	<1%
29	"MultiMedia Modeling", Springer Science and Business Media LLC, 2017 Publication	<1%
30	research.library.mun.ca Internet Source	<1%
31	journals.plos.org Internet Source	<1%
32	www.freepatentsonline.com Internet Source	<1%

33	vuir.vu.edu.au Internet Source	<1%
34	vdocuments.mx Internet Source	<1%
35	Submitted to International Islamic University Chittagong Student Paper	<1%
36	mihaioltean.github.io Internet Source	<1%
37	towardsdatascience.com Internet Source	<1%
38	citeseerx.ist.psu.edu Internet Source	<1%
39	iyfnet.org Internet Source	<1%
40	"Statistical Atlases and Computational Models of the Heart. ACDC and MMWHS Challenges", Springer Science and Business Media LLC, 2018 Publication	<1%
41	Hamed Habibi Aghdam, Elnaz Jahani Heravi. "Guide to Convolutional Neural Networks", Springer Science and Business Media LLC, 2017 Publication	<1%

42	"Major Diseases of Betelvine and Their Management", Crop Diseases and Their Management, 2016. Publication	<1%
43	"Proceedings of the Eighth International Conference on Soft Computing and Pattern Recognition (SoCPaR 2016)", Springer Science and Business Media LLC, 2018 Publication	<1%
44	Shankar A. Deka, Dusan M. Stipanovic, Claire J. Tomlin. "Feedback-Control Based Adversarial Attacks on Recurrent Neural Networks", 2020 59th IEEE Conference on Decision and Control (CDC), 2020 Publication	<1%
45	Souhir Khessiba, Ahmed Ghazi Blaiech, Khaled Ben Khalifa, Asma Ben Abdallah, Mohamed Hédi Bedoui. "Innovative deep learning models for EEG-based vigilance detection", Neural Computing and Applications, 2020 Publication	<1%
46	A. Jeyanthi Suresh, J. Visumathi. "Inception ResNet deep transfer learning model for human action recognition using LSTM", Materials Today: Proceedings, 2020 Publication	<1%
47	P. Adamidis, V. Petridis. "On the parallelization	

of artificial neural networks and genetic algorithms", International Journal of Computer Mathematics, 1998
Publication

<1%

48 "Pattern Recognition and Machine Intelligence", Springer Science and Business Media LLC, 2019
Publication

<1%

Exclude quotes	Off	Exclude matches	Off
Exclude bibliography	Off		