# “Recognition of rice plant diseases using deep learning”

**A Report Submitted in Partial Fulfilment of the Requirements for the End Semester Evaluation of Project-1(EC-498)**

**By**



| Name | Scholar Id |
|---|---|
| Ratul Mazumdar | 1914037 |
| Vaddepudi Nikhil | 1914122 |

**Under the Guidance of**

**Dr. Madhumita Banerjee**

Associate Professor

**Electronics and Communication Engineering Department**

**NATIONAL INSTITUTE OF TECHNOLOGY SILCHAR**

# Contents

# Abstract

The population of the entire planet is growing every day. The main problem in the future years will be to feed everyone on the planet. Rice is one of the most significant crops as it feeds more than 50% of the world's population. Early disease detection is the main difficulty in rice crop cultivation. However, when productivity is affected, it might be challenging to diagnose sickness with the naked eye. Deep learning techniques, in particular convolutional neural networks, are frequently utilised in pattern recognition and machine vision tasks. In order to identify plant diseases, researchers proposed various deep learning models. However, because deep learning models need a lot of parameters, it takes more time to train them and is challenging to use them on small devices.

In this, We taken 10407 training images and 3469 testing images with 10 classes having 9 diseases and one normal class. We segmented the images by K means clustering with the cluster size of 5. We used 3 classifiers -CNN model, Inception Model and VGG-19 model. We have performed the rice disease detection using techniques CNN, VGG-19 and Inception v3. The dataset we used is used for various purpose they have classes of each disease in rice plant in which we perform the experiment and evaluate our result we get the training accuracy of 97.87% in CNN, 99.99% on VGG-19 and 93.48% in Inception v3 with validation score of 75.51% ,89.01% and 77.92% respectively. Our analysis of the experimental data leads us to the conclusion that the VGG-19 model is superior to other classification models for this dataset and provides greater accuracy.

# Introduction

A majority of people all over the world, particularly in Asian nations like India, Japan, China, Thailand, etc., rely on rice as one of their main sources of food. More than half of the world's population eats it frequently as a source of energy. Farmers put a lot of effort and time into preventing diseases, and they utilize their meagre human eye technique to find problems, which results in unseemly agriculture. Indian farmers cherish agricultural land because it helps them create healthy food. The state government-designed method is being used to grow rice in many Indian states. Every year, the Farmers produced a lot of rice. Rice production has recently been impacted by water restrictions, climate change, global warming, and leaf diseases. The rice plant leaf got infected with a wide range of diseases as a result of many sources.

Over the years, rice diseases have consistently impacted both rice output and the agricultural sector. Every time a plant's leaf becomes infected, the virus spreads to all other parts of the plant, lowering the farmer's income.

Some most common diseases like i) Rice bacterial blight is one of the deadly bacterial diseases, which is the most destructive affliction of cultivated rice (Oryza sativa and O. glaberrima). It is also known as the bacterial blight of rice. ii) Blast, also called rotten neck, is one of the most destructive diseases of Missouri rice. Losses due to this disease have been on the increase since 2000. iii) Brown spot is the one of the major fungal diseases in rice in which it is caused by Bipolarisoryzae. This disease mainly attacks the crop from seedling stage to milky stage. Brown spot causes both quality and quantity losses. iv) Rice hispa is generally known to attack young rice plants, but in epidemic situations it invades mature plants as well. Heavy infestations in outbreak-prone areas starts in December on the local *Boro* and continues upto the following November on transplanted *Aman* crops.

Diseases cause a loss in rice output of about 10%, which might increase to 30–40% and even result in a single poor crop, which can be disastrous for the farmers involved. Climate change, water scarcity, plus leaf diseases have almost all recently had an effect on rice production. The leaf of the rice plant developed a number of diseases as a result of numerous sources.

Diseases of rice plant leaves have an effect on the yield of the entire rice crop. Deep learning algorithms in agriculture enable the farmer to monitor and care for each plant separately. Statistics, detection of diseases, crop categorization, and control are all possible with deep learning. In a different way, the deep learning method directly takes the crucial data from the image collection. Deep learning uses neurons to mimic human thinking. Deep Learning is constructed using a multi-neural network configuration with numerous hidden layers to predict the required outputs. The domain of image detection has made extensive use of neural networks (CNN), one of the network designs included in deep learning.

# Literature Review:

| Authors | Paper title | Details |
|---|---|---|
| Shital Pawar[1][2022] | Leaf Disease Detection of Multiple Plants Using Deep Learning | • In this 15 layer CNN is used to detect the disease and suggested the pesticides based on the disease found .<br>• This proposed model can detect diseases in multiple plants by simply adding a set of crop and training data<br>This model uses Gaussian size limited filter to pre scan the image to supress the leaf area and send it to the classifier. |
| Santosh Kumar Upadhyay [2][2022] | Deep Transfer Learning-Based Rice Leaves Disease Diagnosis and Classification model using InceptionV3 | • The model used a Deep CNN and an Inception v3 Transfer learning model to identify rice disease.<br>• The pre-trained Inception v3 model is used to add weight to the data, which is subsequently translated into the experiments for feature extraction using the transfer learning model. |
| SK Mahmudul Hassan[3][2022] | Plant Disease Identification using a Convolutional Neural Network | • A new CNN architecture is proposed using Inception and Residual connection, which extracts better features and produces higher performance results.<br>• The proposed architecture uses fewer parameters than other deep learning architecture and is also faster than the standard deep learning models. |

| | | |
|---|---|---|
| Naresh Cherukuri [4][2021] | Automated Classification of rice leaf disease using Deep Learning Approach | • In this paper proposed a Artificial Neural Network model and a KNN model to detect and classify 5 different diseases.<br>• The essential features for the classification of the disease are extracted from the images taken from the segmented image and given as input for ANN |
| Amratya Bhattachrya[5][2021] | A Novel Deep Learning Based Model for Classification of Rice Leaf Diseases | • In this transfer learning based model is used for classification of 5 different classes of diseases.<br>• The weights from pre trained network has been set but the layers has been trained from scratch. |
| Ahmed Faizal[6][2020] | Smartphone Application for Deep Learning-Based Rice Plant Disease Detection | • For rice plant disease detection, they developed an application in deep learning on cloud server and application on smartphone<br>• They used VGG16 architecture for detection with a test accuracy of 60%. |

# Technical Novelty

In this proposed system, we have taken a complete different dataset from any other available datasets in the market.

This model can be used in multiple fields such as detection of plant diseases by using drones because this system has the ability to recognize plant diseases even from images having the plants at certain distance.

In this proposed work, we have used VGG-19 model for the classification of images in order to get better accuracy.

# Methodology

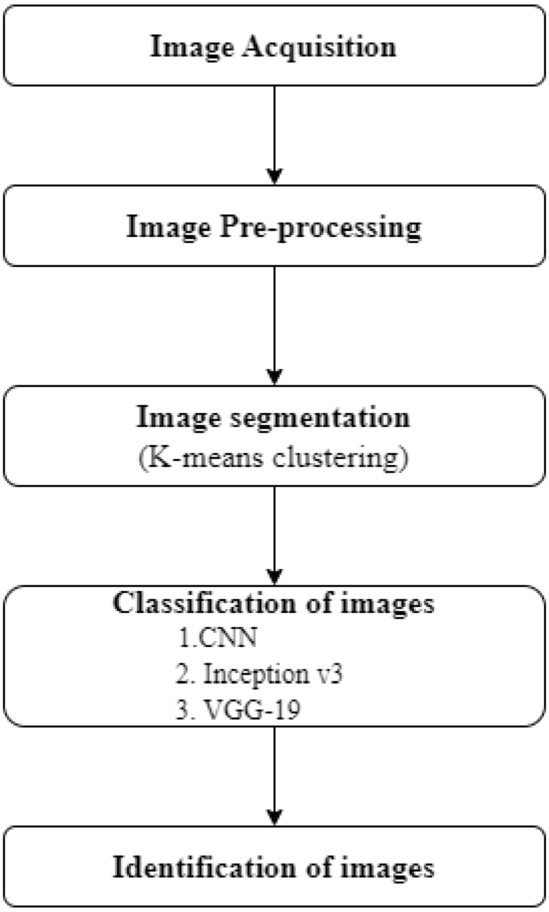



**fig:** flow chart for proposed work

**Image Acquisition** :- A total of 10407 training images and 3469 testing images were collected from different agricultural websites and paddy fields [5]. These images are captured in 480×640 pixels.

**Table-1:** Image Collection of different classes.

| Class Name | No. of Images |
|---|---|
| **Bacterial Leaf Blight** | 479 |
| **Bacterial leaf streak** | 380 |
| **Bacterial panicle blight** | 337 |
| **Blast** | 1738 |
| **Brown Spot** | 965 |
| **Dead heart** | 1442 |
| **Downy mildew** | 620 |
| **Hispa** | 1594 |
| **Normal** | 1764 |
| **Tungro** | 1088 |

1. Bacterial Leaf Blight

2. Bacterial leaf streak

3. Bacterial panicle blight

4. Blast

5. Brown Spot

6. Dead heart

7. Downy mildew

8. Hispa

9. Normal

10. Tungro

**Fig-1:** Rice plants having different types of diseases

**Image Pre-Processing :-**

i. **Converting RGB Images Into Gray Scale:**

The RGB images are taken and we need to convert these images to Gray scale. The main idea behind this is to get display direction densely.

**Fig-2(a):** RGB Image **Fig-2(b):** Gray Scale

**Image Segmentation: -**

It employs to reduce image's complexity and simplify it**.**

A **K-means clustering technique** has been engaged in this. K means clustering partitions or makes the clusters of the data which is nearest to the centroid. Centroid is arithmetic mean of the all cluster points. The new centroid is estimated for each of the partition or cluster, after allocating the nearest centroid. The position and number of the centroid is changed step by step till k-clusters are derived and no more changes are possible.

The crucial part of this technique is selecting the K values. The values will be taken by using trial and error method.

Considering first set of k means is $M_1, M_2, \ldots\ldots M_k$, and the algorithm is done in two alternative methods:

Assignment Method: Assign each consideration to the cluster whose mean produces sum of squares lowest within the cluster, where sum of squares is probably the nearest mean.

$$S_i^t = \left[X_p : \left(X_p - M_i^t\right)^2 \leq \left(X_p - M_j^t\right)^2 \ldots . \forall j, 1 \leq j \leq k\right] \qquad (1)$$

where each $X_p$ is allocated to explicitly one $S^t$ , even if it could be allotted to two or more of them.

Update Method: After observation, the new means as the centroid of the new clusters can be calculated as,

$$M_j^{t+1} = \frac{1}{|S_j^t|} \sum_{X j \in S_j^t} X_j \quad (2)$$

**Fig-3(a):** Gray scale image

**Fig-3(b):** Segmented image

## Classification of Images:

We divided the segmented image dataset into 3 categories:

1. Training set: 70% images (7286 images) which is for training
2. Validation set: 10% images (1037 images) which is for validation for each epoch
3. Testing set: 20% images (2093 images) which is for testing

For classification we used 3 multi class classifiers as follows:

1. Convolutional Neural Network (CNN)
2. Inception V3
3. VGG-19

### Convolutional Neural network (CNN):

Convolutional neural networks (CNNs) are a particular kind of neural network that are exceptionally good at recognising patterns and classifying images in computer vision applications. CNN has the advantage of being able to learn from training photos and automatically extract features, whereas the old approach requires manually extracting features from the images. Convolutional, pooling, and fully connected layers form the CNN architecture.

he fundamental and most important part of CNN that extracts the information from the input image is the convolutional layer. Convolutional layers generate feature maps as their output by applying a tiny array of numbers referred as kernels to the input. Various convolutional kernels are employed to extract various feature types. Convolutional layers vary in number according to the shape of input images.

The convolutional feature map's dimension is decreased through pooling, which is done after the convolutional layer. By reducing the size of the feature map, the pooling layer executes down sampling operations, which ultimately assists in lowering the needed computational complexity to analyse the input. There are various pooling operations, including max-pooling, min-pooling, and average-pooling. The convolutional or pooling layer's output feature maps are converted into a single-dimensional vector in which each input and each output are associated by weight. A dense layer is another name for this one. One or more fully connected levels are possible, and the number of outputs on the last fully linked layer corresponds to the number of classes.

The following is the proposed CNN model we used in this project:

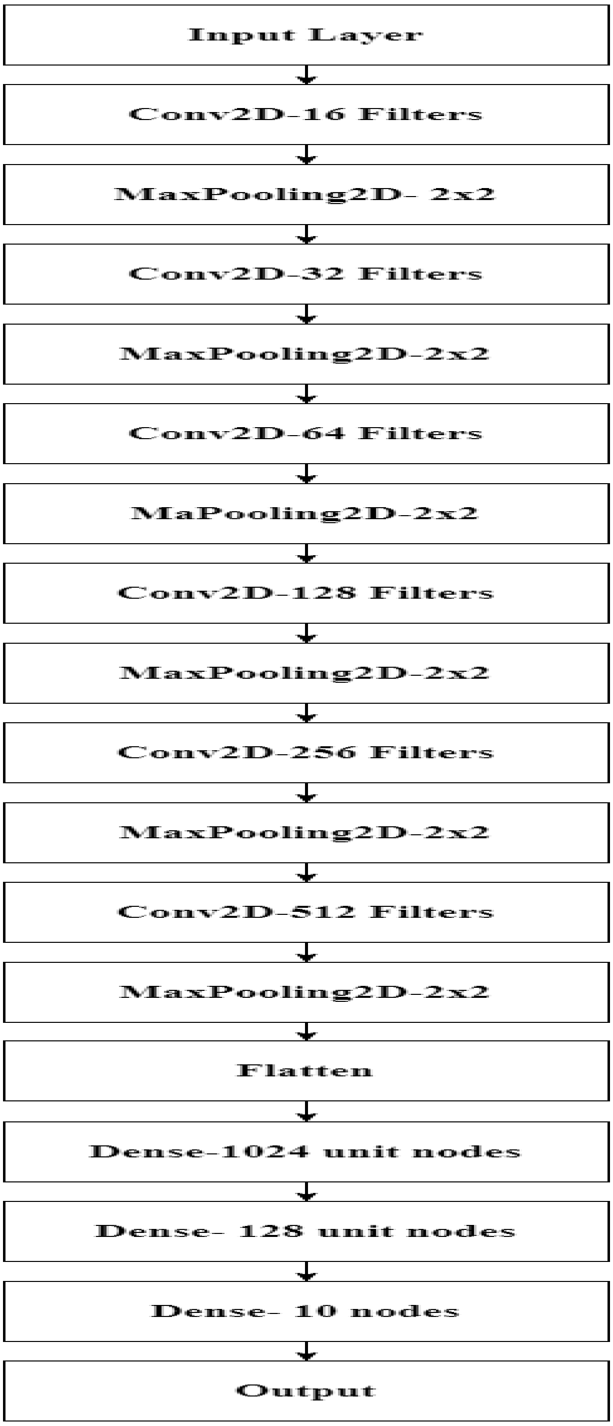



**Fig-4:** proposed CNN model

## Inception v3:

The Inception V3 is a deep learning model based on Convolutional Neural Networks, which is used for image classification. On the ImageNet dataset, this model has been demonstrated to achieve higher than 78.1% accuracy. The model is the result of numerous concepts that have been established by various researchers over the decades. There are 42 layers in the inception V3 model.

Convolutions, average pooling, max pooling, concatenations, dropouts, and fully linked layers are some of the symmetric and asymmetric building components that make up the model itself. The model makes considerable use of batch normalisation, which is also applied to the activation inputs. SoftMax is employed to calculate loss.

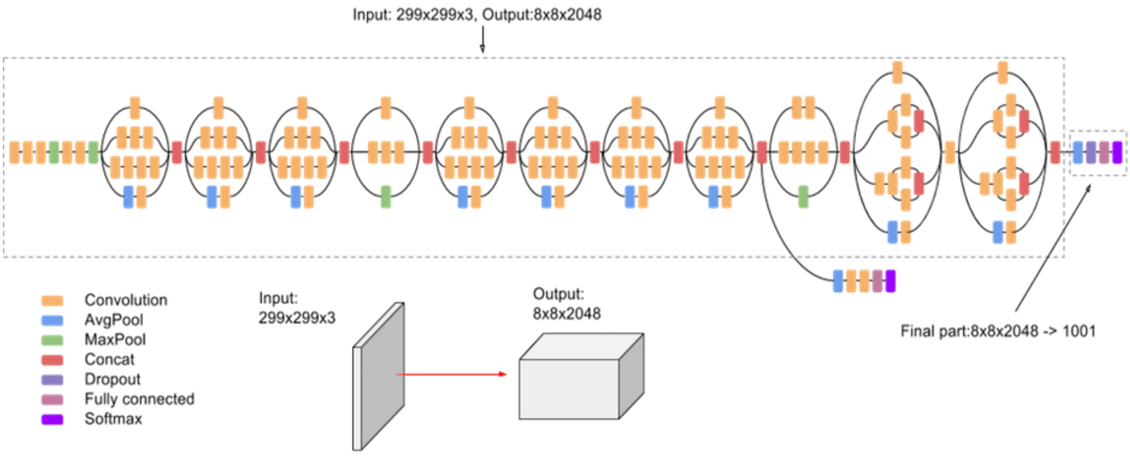



**Fig-5:** Architecture of Inception model

In this project, we ignored fully connected layers, batch normalised after the last concatenated layer and flattened it, and added a dense layer of 128 unit nodes and one dropout layer of 20%. Then we added one dense layer of 10 unit nodes, and we got the results as follows:

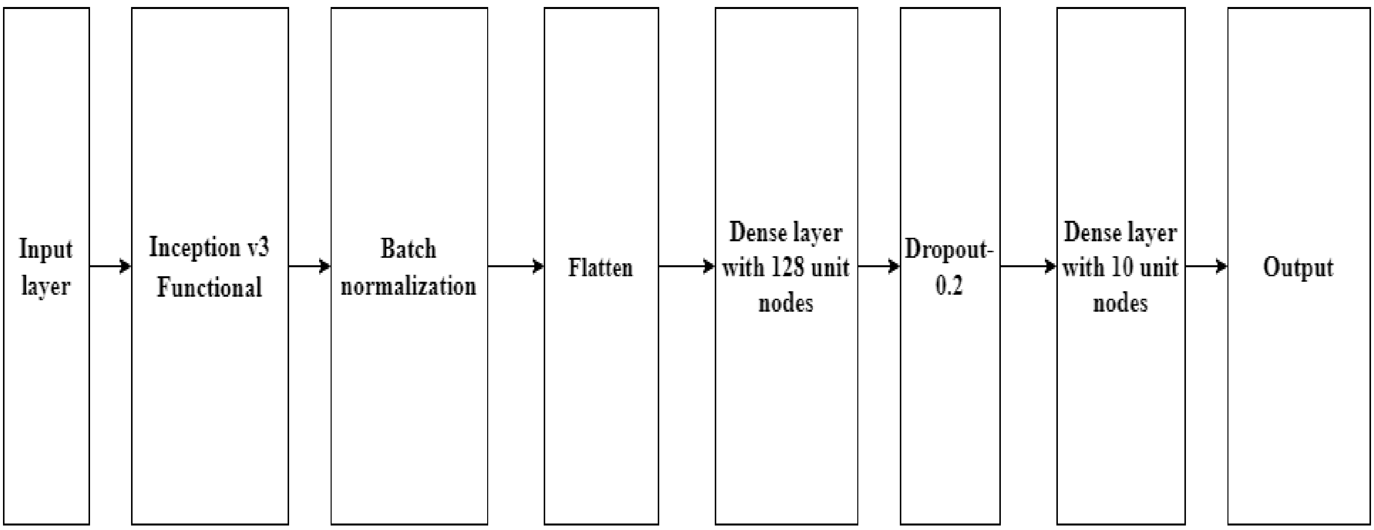



**Fig-6:** Proposed Inception v3 Model

### VGG-19:

**VGG19** stands for Visual Geometry Group. It is a convolutional neural network architecture that is 19 layers deep. It is a pre-defined model which is used to process data in such a way that it will extract all the essential features.

The architecture of VGG19.

- **Input:** The VGG19 takes in an image input size of 224×224.
- **Convolutional Layers**: VGG's convolutional layers leverage a minimal receptive field, i.e., 3×3, the smallest possible size that still captures up/down and left/right. This is followed by a **ReLU** activation function. **ReLU** stands for rectified linear unit activation function, it is a piecewise linear function that will output the input if positive otherwise, the output is zero. Stride is fixed at 1 pixel to keep the spatial resolution preserved after convolution
- **Fully-Connected Layers:** The VGG19 has 3 **fully connected** layers. Out of the 3 layers, the first 2 have 4096 nodes each, and the third has 1000 nodes, which is the total number of classes the **ImageNet** dataset has.

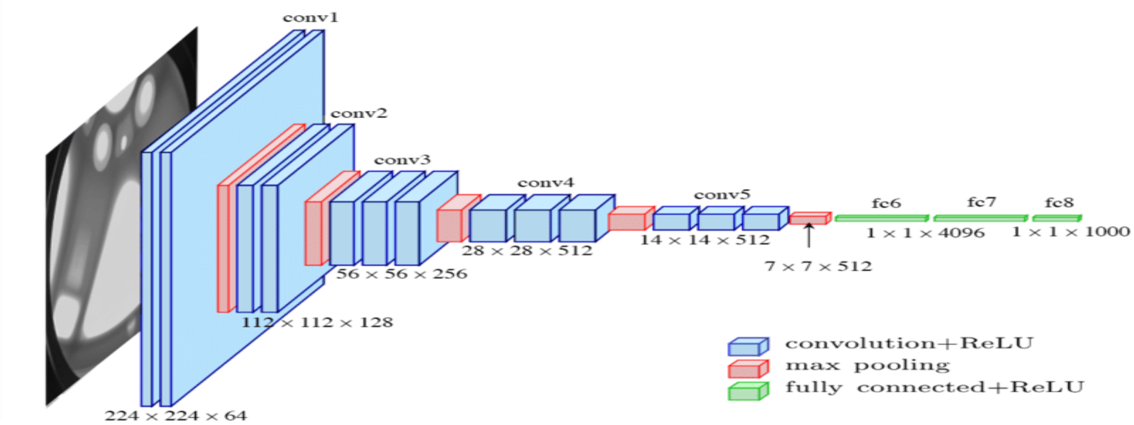



**Fig-7:** Architecture of VGG-19 model

In this project used vgg-19 model without including the fully connected layers and we modified it by flatten it after last max pooling layer and added one Dropout layer with 9% and Dense layer with 10 unit nodes with activation SoftMax because our dataset is having 10 classes to classify.

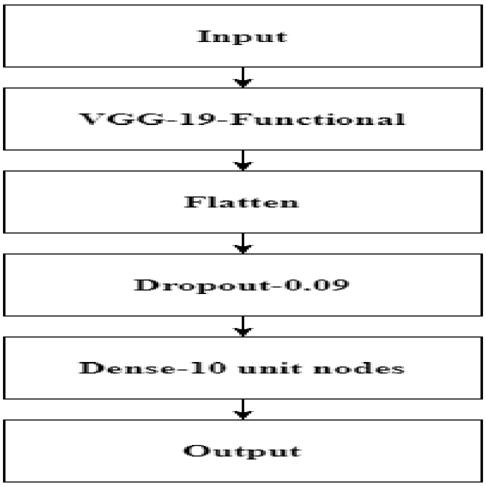


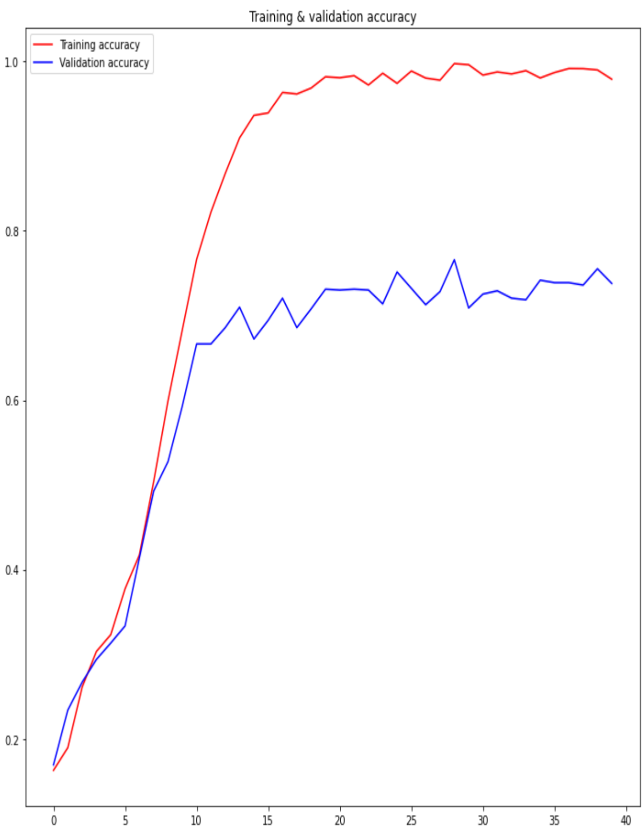

**Fig-8:** Proposed VGG-19 Model

# Results

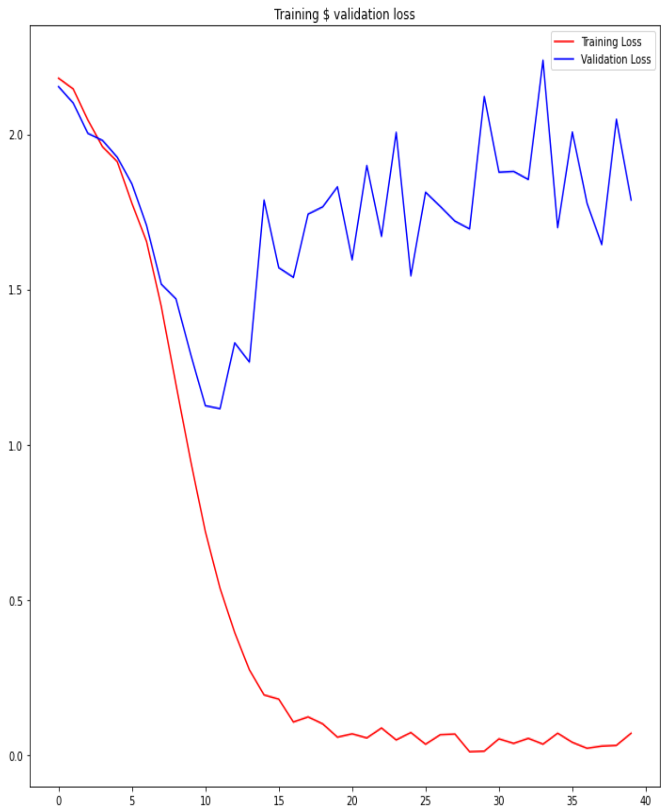
After segmentation, we split segmented images into three parts:

1. Training set: 70% images (7286 images) which is for training
2. Validation set: 10% images (1037 images) which is for validation for each epoch
3. Testing set: 20% images (2093 images) which is for testing

First, as shown in Figure 4, we used the **proposed CNN model**. We are given a batch size of 64 images and a total of 40 epochs. The training loss was 0.0714 and the training accuracy wa s 97.87%. Validation loss is 1.6447, and validation accuracy is 75.51%. The testing loss was 1 .6624 and the testing accuracy was 75.63%.
Graphs between Training and validations losses and accuracy as follows:




a)




b)

**Fig-9:** a) Training accuracy vs validation accuracy graph to the proposed cnn model

b) Training loss vs validation loss graph to the proposed cnn model

As per the figure-9b we can tell that model is getting over-fitting. The graph of validation loss is increasing after 13 epoch. The confusion matrix of proposed cnn model is given below.

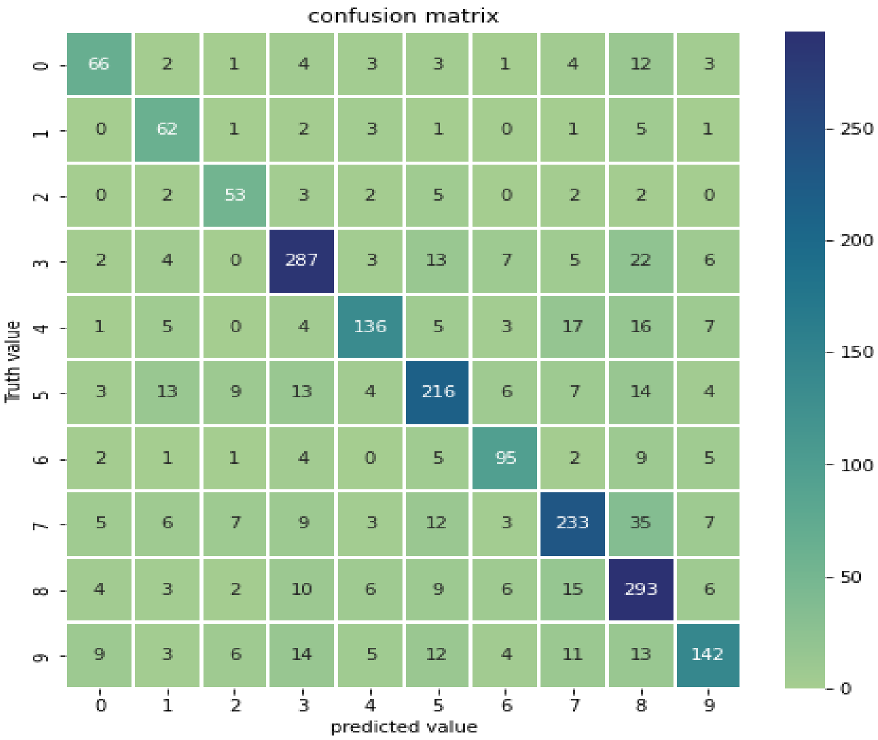
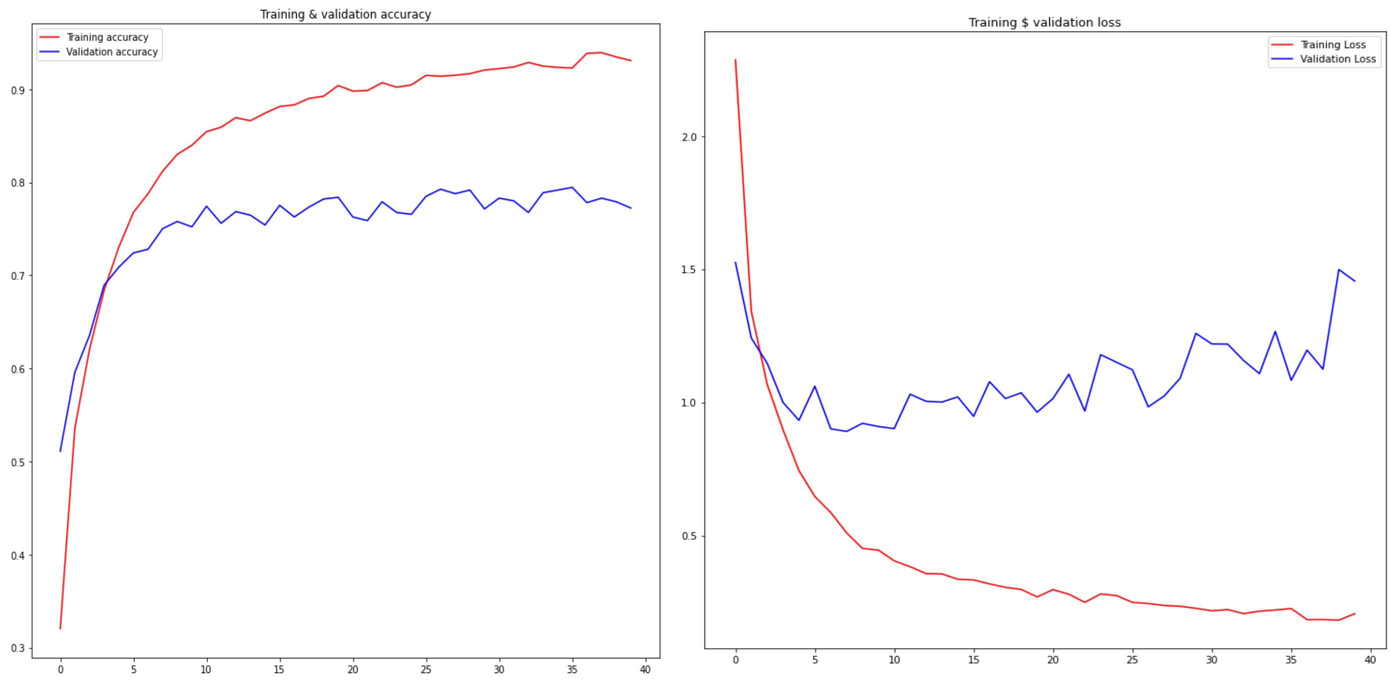



**Fig-10:** confusion matrix of proposed cnn model

As shown in Figure 6, we used the **proposed Inception model**. We are given a batch size of 6 4 images and a total of 40 epochs. The training loss was 0.01818 and the training accuracy wa s 93.48%. Validation loss is 1.4990, and validation accuracy is 77.92%. The testing loss was 1 .5529 and the testing accuracy was 78.59%.

The graphs between training and validation losses and accuracies:




**a)** **b)**

**Fig-11:** a) Training accuracy vs validation accuracy graph to the proposed Inception model

b) Training loss vs validation loss graph to the proposed Inception model

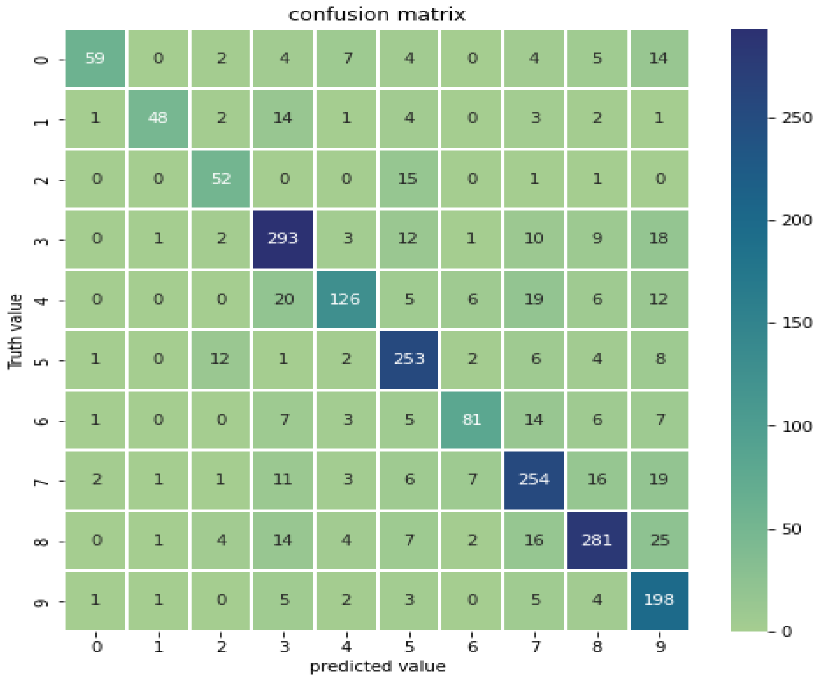
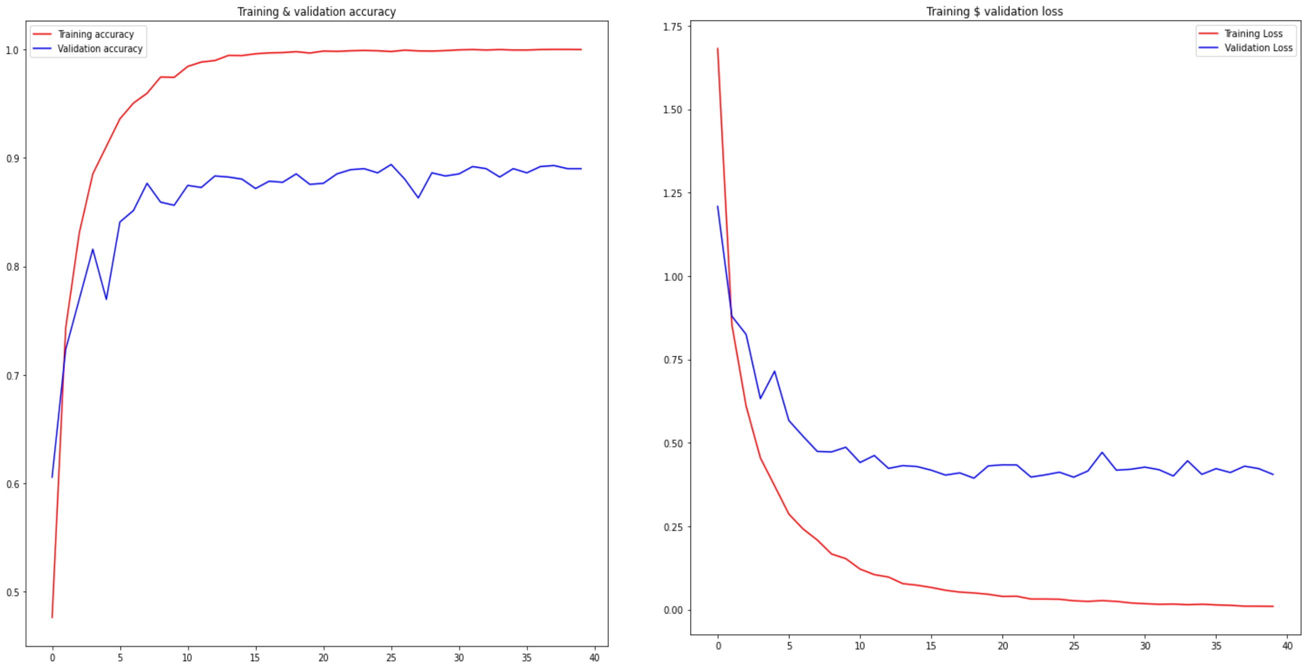



**Fig-12:** Confusion matrix of proposed Inception Model

As shown in Figure 8, we used the **proposed VGG-19 model**. We are given a batch size of 64 images and a total of 40 epochs. The training loss was 0.0098 and the training accuracy was 99.999%. Validation loss is 0.4053, and validation accuracy is 89.01%. The testing loss was 0.4247 and the testing accuracy was 88.91%.

The graphs between training and validation losses and accuracies:




a) b)

**Fig-13:** a) Training accuracy vs validation accuracy graph to the proposed VGG-19 model

b) Training loss vs validation loss graph to the proposed VGG-19 model

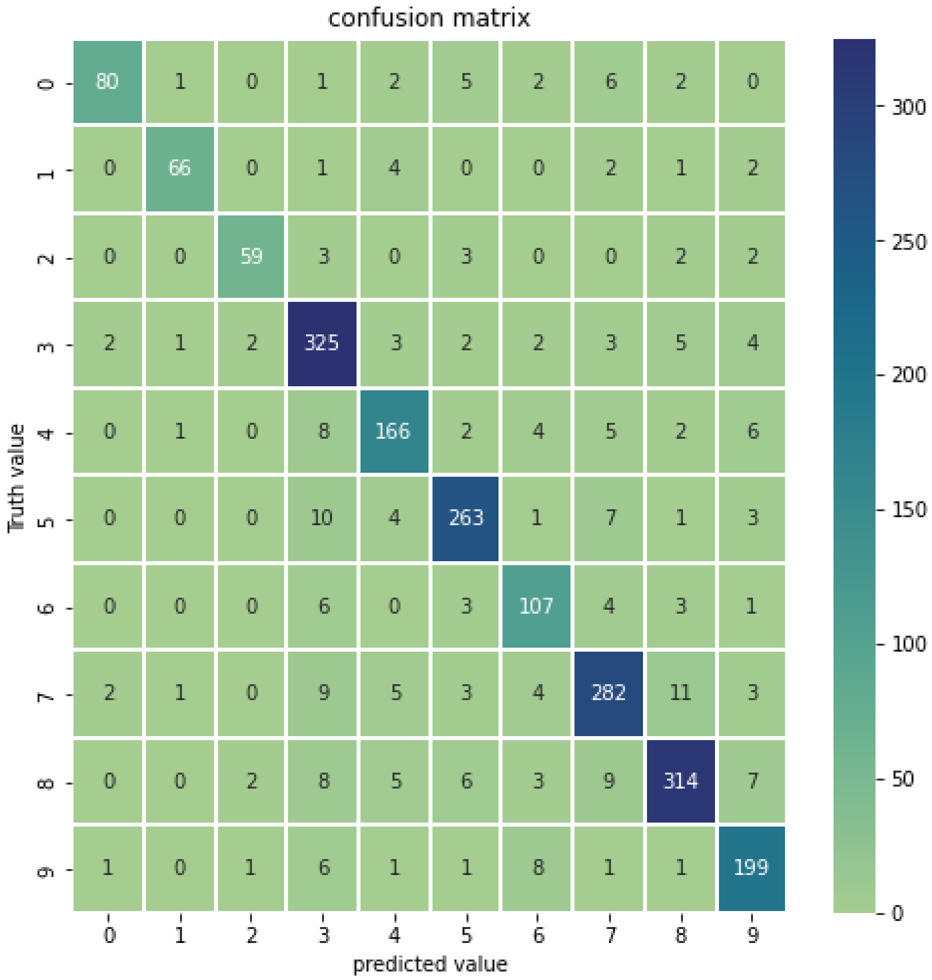



**Fig-14:** Confusion matrix of proposed VGG-19 model

The comparison among the proposed cnn, Inception, VGG-19 model as follows:

| | CNN model | Inception model | VGG-19 model |
|---|---|---|---|
| **Accuracy:** | | | |
| Training | 97.87% | 93.48% | 99.99% |
| Validation | 75.51% | 77.92% | 89.01% |
| Testing | 75.63% | 78.59% | 88.91% |
| **Loss:** | | | |
| Training | 0.0714 | 0.01818 | 0.0098 |
| Validation | 1.6447 | 1.4990 | 0.4053 |
| Testing | 1.6624 | 1.5529 | 0.4247 |

**Table-2:** Comparison among models

**Table-3:** Precisions of classes with different classifier models

| Classes | CNN model precision | Inception model precision | VGG-19 model precision |
|---|---|---|---|
| Bacterial leaf Blight | 0.72 | 0.91 | 0.94 |
| Bacterial leaf streak | 0.61 | 0.92 | 0.94 |
| Bacterial panicle blight | 0.66 | 0.69 | 0.92 |
| Blast | 0.82 | 0.79 | 0.86 |
| Brown spot | 0.82 | 0.83 | 0.87 |
| Dead heart | 0.77 | 0.81 | 0.91 |
| Downy mildew | 0.76 | 0.82 | 0.82 |
| Hispa | 0.78 | 0.77 | 0.88 |
| Normal | 0.70 | 0.84 | 0.92 |
| tungro | 0.78 | 0.66 | 0.88 |
| **Total:** | 0.76 | 0.79 | 0.89 |

The Roc graphs of different classifiers:

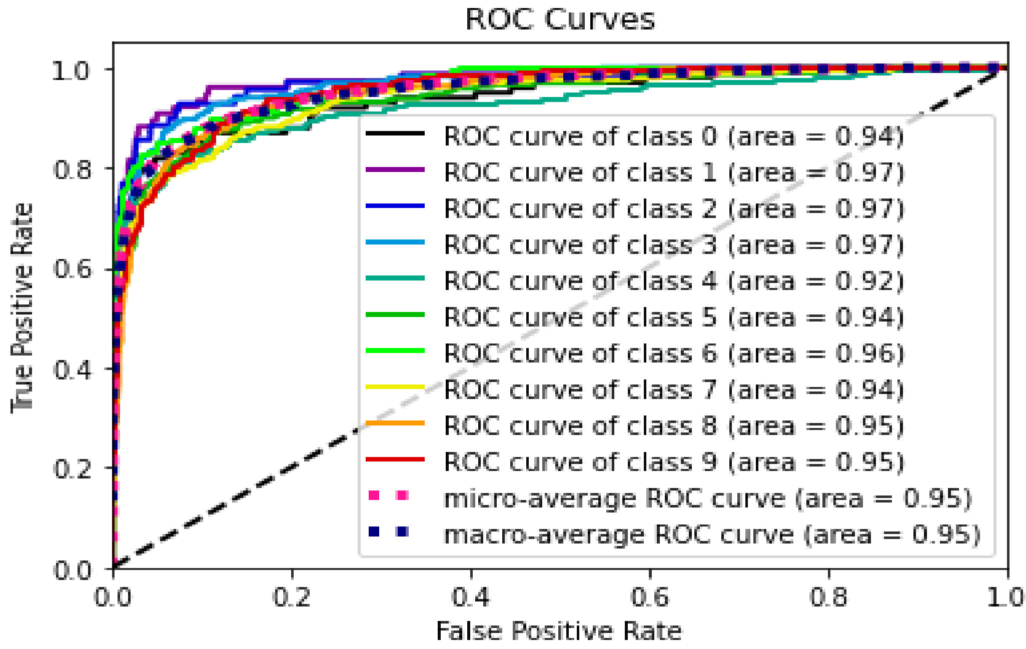



**Fig-15:** Roc curve for Proposed CNN model

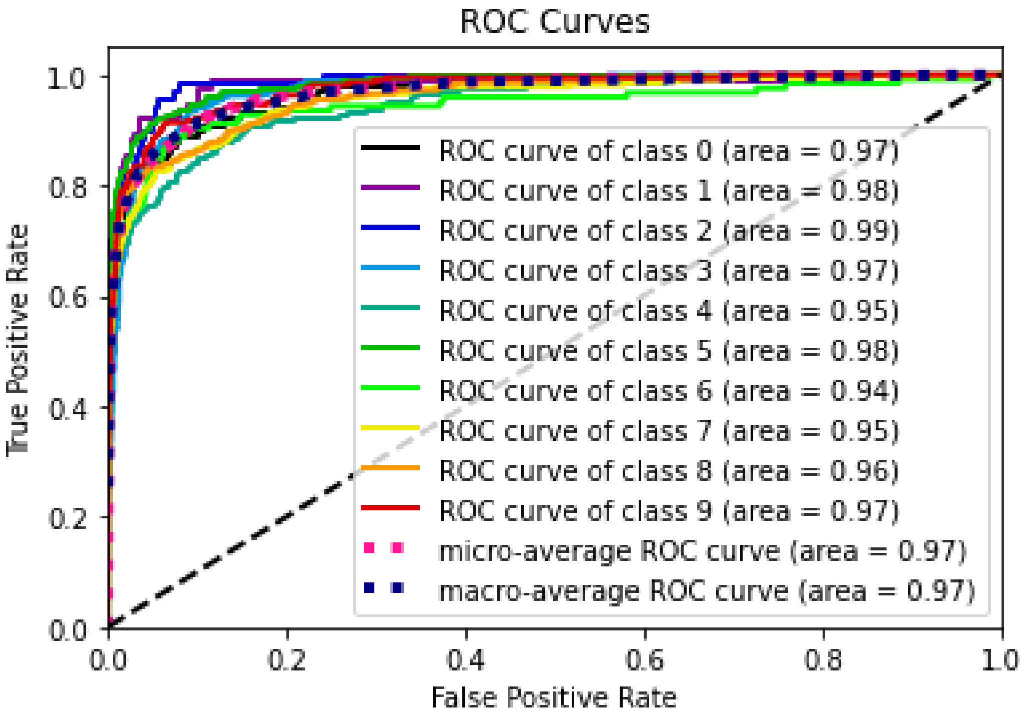



**Fig-15:** Roc curve for Proposed Inception model

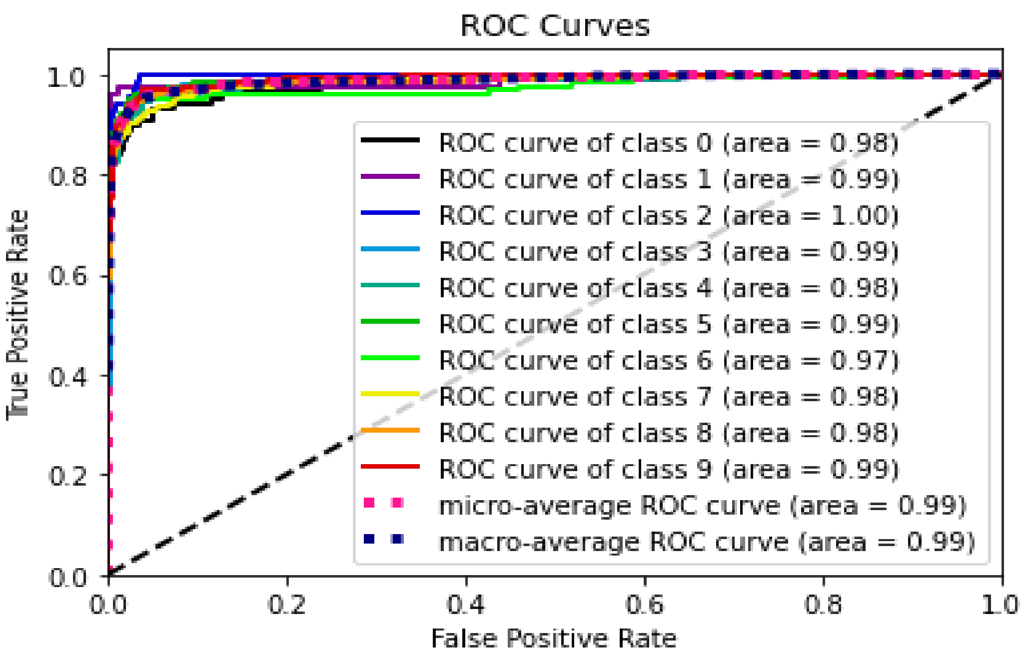



**Fig-15:** Roc curve for Proposed VGG-19 model

# Conclusion

In this project, we have performed the rice disease detection using techniques CNN, VGG-19 and Inception v3. The dataset we used is used for various purpose they have classes of each disease in rice plant in which we perform the experiment and evaluate our result we get the training accuracy of 97.87% in CNN, 99.99% on VGG-19 and 93.48% in Inception v3 with validation score of 75.51% ,89.01% and 77.92% respectively.

The segmentation is done by using the k-means clustering algorithm, which provides higher accuracy with less computational time.

From the results and comparison, we can declare that VGG-19 model is best in detection of rice diseases than remaining models of classification for this dataset.

This will help the farmer to take preventive measures before the disease gets spread in entire farm. It will help in increase the production of rice.

# Future Work

- We will increase dataset size with adding some images from different resources
- We will increase accuracy by doing following functions:
  - Augmentation of Images
  - Different multi-class classifiers like MobileNet and few more.
- Create an application of detection of Disease without clicking image or snapping a image.

# References


[1] Sammy V. Militante, Bobby D. Gerardo, Nanette V. Dionisio "Leaf Disease Detecton of Multiple Plants using deep learning" *2019 IEEE Eurasia Conference on IOT, Communication and Engineering* (ECICE)

[2] Rukhsar Santosh Kumar Upadhyay, "Deep Transfer Learning-Based Rice Leaves Disease Diagnosis and Classification model using InceptionV3 "*2022 International Conference on Computational Intelligence and Sustainable Engineering Solutions (CISES)*

[3] Sk Mahmudul Hassan, Arnab Kumar Maji," Plant Disease Identification using a Convolutional Neural Network " IEEE Access volume 10, edition 2022

[4] Naresh Cherukuri , G.Ravi Kumar, Ongole Gandhi, Dr.Venkata Sai Krishna Thotakura Dama NagaMani, CMAK Zeelan Basha "Automated Classification of rice leaf disease using Deep Learning Approach" *2021 5th International Conference on Electronics, Communication and Aerospace Technology (ICECA)*

*[5]* Amartya Bhattacharya "A Novel Deep Learning Based Model for classification of Rice Leaf Diseases " *2021 Swedish Workshop on Data Science (SweDS)*

[6] Heri Andrianto , Suhardi, Ahmad Faizal , Fladio Armandika "SmartPhone Application for Deep Learning Based Rice Plant Disease Detection" 2020 International Conference on Information Technology Systems and Innovation (ICITSI)

[7] Ministry of rural development : https://rural.nic.in/

[8] Farmer's portal : https://farmer.gov.in/