PREDICTING RED ROT DISEASE IN SUGARCANE LEAVES USING DEEP LEARNING TECHNIQUES

Project Submitted to the
SRM University AP, Andhra Pradesh
for the partial fulfillment of the requirements to award the degree of

Bachelor of Technology
in
Computer Science & Engineering
School of Engineering & Sciences

submitted by

Perumalla Hemanth - AP20110010019

Morthala Venkata Sai Nagarjuna Reddy - AP20110010026

Vedanabhatla V M P Surya Sai Harshith - AP20110010045
Panyala Sandeep Reddy - AP20110010064

Under the Guidance of
Dr. Satish Anamalamudi

Department of Computer Science & Engineering
SRM University-AP
Neerukonda, Mangalgiri, Guntur
Andhra Pradesh - 522 240

[May, 2024]

DECLARATION

I undersigned hereby declare that the project report **PREDICTING RED ROT DISEASE IN SUGARCANE LEAVES USING DEEP LEARNING TECHNIQUES** submitted for partial fulfillment of the requirements for the award of degree of Bachelor of Technology in the Computer Science & Engineering, SRM University-AP, is a bonafide work done by me under supervision of Dr. Satish Anamalamudi. This submission represents my ideas in my own words and where ideas or words of others have been included, I have adequately and accurately cited and referenced the original sources. I also declare that I have adhered to ethics of academic honesty and integrity and have not misrepresented or fabricated any data or idea or fact or source in my submission. I understand that any violation of the above will be a cause for disciplinary action by the institute and/or the University and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been obtained. This report has not been previously formed the basis for the award of any degree of any other University.

Place	: Guntur	Date	: May 12, 2024
Name of student	: Perumalla Hemanth	Signature	: P. Hemanth.
Name of student	: Morthala Venkata Sai Nagarjuna Reddy	Signature	: M. Nagarjuna
Name of student	: Vedanabhatla V M P Surya Sai Harshith	Signature	: V Sai Harshith
Name of student	: Panyala Sandeep Reddy	Signature	: P. Sandeep Reddy

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

SRM University-AP

Neerukonda, Mangalgiri, Guntur

Andhra Pradesh - 522 240

CERTIFICATE

This is to certify that the report entitled **PREDICTING RED ROT DISEASE IN SUGARCANE LEAVES USING DEEP LEARNING TECHNIQUES** submitted by **Perumalla Hemanth , Morthala Venkata Sai Nagarjuna Reddy , Vedanabhatla V M P Surya Sai Harshith , Panyala Sandeep Reddy** to the SRM University-AP in partial fulfillment of the requirements for the award of the Degree of Bachelor of Technology in Computer Science & Engineering is a bonafide record of the project work carried out under our guidance and supervision. This report in any form has not been submitted to any other University or Institute for any purpose.

Project Guide	Head of Department
Name : Dr. Satish Anamalamudi	Name : Dr. Niraj Upadhyaya
Signature:	Signature:

ACKNOWLEDGMENT

We would like to convey our sincere tribute to everyone who helped us to complete this Project Report titled **PREDICTING RED ROT DISEASE IN SUGARCANE LEAVES USING DEEP LEARNING TECHNIQUES** and present it satisfactorily.

Especially we would like to thankful for our guide and supervisor Dr. Satish Anamalamudi in the Department of Computer Science & Engineering for giving me valuable suggestions and critical inputs in the preparation of this report. Our supervisor, deserves special thanks for his help, steering the team toward the goal and pushing us to keep moving forward and support us throughout the process and the writing of this report. We also appreciate the time you took to proofread and rectify our numerous mistakes. We are also thankful to Dr. Niraj Upadhyaya, Head of Department of Computer Science & Engineering for encouragement.

Perumalla Hemanth , Morthala Venkata Sai Nagarjuna Reddy ,
Vedanabhatla V M P Surya Sai Harshith , Panyala Sandeep Reddy
(Reg. No. AP20110010019, AP20110010026, AP20110010045, AP20110010064)
B. Tech.
Department of Computer Science & Engineering
SRM University-AP

ABSTRACT

One of the most important crops in India is Sugarcane, which is the second highest producer after Brazil in the world. But one of the leaf diseases that affects the sugarcane yield is Red Rot. It must be found prior to effectively survive the illness and protect the crop. In this study, we will be training customized models using the CNN architecture to identify the red rot disease. By implementing VGG16, ResNet50, MobileNet, Inception V3, and a hybrid model for training them with the image dataset along with Adam's, SGD (Stochastic Gradient Descent) and RMSProp (Root Mean Square Proportion) optimizers for enhancing the model's performance. We compare the performance of each model and optimizer combination by means of in-depth testing and analysis. The combination that best predicts the occurrence of red rot disease in sugarcane plants is what we want to identify. By helping to develop effective and trustworthy methods for early disease detection in agriculture, this research eventually helps farmers preserve the production and health of their crops

Keywords: CNN, Transfer Learning, Adam's Optimizer, SGD Optimizer, RMSProp optimizer, Hybrid model, MobileNet, V3 Inception, ResNet50, VGG16.

CONTENTS

LIST OF TABLES

LIST OF FIGURES

Chapter 1

INTRODUCTION

In India, Agriculture is the most important division that forms a circle around the cultivation of crops, in which Sugarcane is one of the important crops. Sugarcane has been there for 10,000 years. It is first cultivated in the Polynesia region which is made up of over 1000 islands in the Pacific ocean and around the period of 1400-1000 BCE. Sugarcane is an important crop for many countries like Brazil, it's the world's largest producer of sugarcane and India stands as the second largest producer of sugarcane worldwide. Sugarcane is a perennial crop, meaning it can regrow after each yield.

Sugarcane is primarily used for the mass production of sugars and many other by-products. Sugar is used as a sweetening agent in the foods and beverages we consume daily. Sugars are produced by processing the plant into raw sugar at the processing plant near the cane fields and these raw sugars get transported to refineries for refined sugars.

The farmers who are cultivating the sugarcane crop are facing some difficulties due to the decrease in crop's yield. The crop yield gets affected by the following scenarios: Climatic changes like irregular rainfall, increased droughts and greater frequency of extreme weather events, can result in crop losses. Weed infestation - spread of invasive weeds clashes with the sugarcane plant development and yield, by competing for nutrients, water and sunlight. Weather conditions - Droughts, floods, extreme temperature, hailstorms will play a role in the minimization or destruction of sugarcane agricultural produce. Soil Degradation - Poor soil health, erosion, salinity, acidity and nutrient deficiency these all will have a negative impact on the yield of sugarcane. Although, the crop is often plagued by various pests, diseases and the environmental factors which will directly show a major impact on the crop yield and on the quality. With in these trouble factors, Red Rot disease is one primary concern. It will make the farmers affected financially for the loss of the crop and will be a slow fall in the industry graph which are dependents on sugarcane by-products.

Red Rot disease is a considerable problem during the cultivation of sugarcane. It causes major loss to the farmers who cultivate the crop and the industry who gets affected due to decrease in the production of sugarcane. This disease is identified on the leaves of sugarcane. It appears as red color lesions on the leaves. Later, these turn on to brown and dried up, eventually leading the plant to death. This disease is transmittable and can spread at high rate. So to avoid the crop being affected by the red rot disease, its very essential to detect this fungal disease at an early stage to prevent from spreading the disease, which leads to higher crop yield and helps in minimizing the crop loss.

If the crop shouldn't be affected by the red rot disease, early detection of the disease is essential for productive management and control of the disease. There are traditional ways for the identification of red rot disease, by the visual inspection by skillful professionals and laboratory testing. These methods are generally time-consuming, and the results might be often inaccurate. Therefore, we need to create a solution for identifying the disease in a rapid manner, precise detection and cost efficient method in sugarcane plants.

In the current evolving world, the technology has been advanced and every field is getting the work done through technical equipment's. Currently Artificial Intelligence and Machine Learning have taken advanced in the current tech world. Many deep learning techniques are introduced which are considered primarily and a standard approach for image-oriented disease detection in plants. The architecture for the image classification used is CNN – Convolutional Neural Network. CNN has presented with a high success rate in detecting the diseases in various crops which includes the sugarcane crop too. CNN is capable of learning complex patterns and a very detail information from the image datasets which makes them similar for image-based disease detection. On training the image data with CNN, results in achieving high accuracy even though if the data is limited.

Detecting the diseases through the deep learning techniques can benefit in several ways. Primarily, it will authorize a fast and accurate detection of diseases. This will help acknowledging the farmers regarding the disease and allows them to take the required action and preventing the transfer of the disease. Secondly, it will reduce the time for depending on the expertise for the visual detection which might be inaccurate. It can be easily accessible by anyone, mainly the farmers in the rural areas. At last, it's a cost-efficient solution for these disease detections, which helps in reducing the economic burden on the farmers.

1.1 MOTIVATION

Consider a farmer, who has been cultivating sugarcane crop for several years. One day, he had noticed that some of his sugarcane crop leaves are showing the signs of red rot disease. It appears to be in red, yellowish-red lesions on the leaves. The farmer understood that if he doesn't take an action quickly, this disease could spread leading to significant damage for the crop. Meanwhile, the traditional methods for the red rot disease detection are prolonged, labour intensive and the requirement of highly skilful professional.

The motivation behind this project is to challenge the traditional methods in terms of financial, time and to develop a model through the deep learning-based method for the identification of red rot disease in the leaves of the sugarcane crop, which provides a fast, accurate and a budget friendly approach in the detection of the disease. This approach can benefit the farmers in protecting their crops, improvement in the yields leading to reducing the financial loss, which leads to the development of supportable and strong systems for the agriculture. Through the CNN's and deep learning models which contribute for higher accuracy rate in image detection and by training the models with large datasets, can contribute for the improvement in accuracy and efficiency of the red rot disease detection, making them ideal for the real-world applications in agriculture.

1.2 PROBLEM STATEMENT

The problem statement for this project is to develop an accurate and efficient deep learning-based method for detecting Red Rot disease in sugarcane leaves. The current methods for detecting Red Rot disease in sugarcane leaves are time-consuming, labour-intensive, and require specialized expertise. These methods include visual inspection, laboratory testing, and molecular techniques. However, these methods have limitations, such as low accuracy, high cost, and limited availability in resource-poor settings. Therefore, there is a need for a more accurate, efficient, and accessible method for detecting Red Rot disease in sugarcane leaves.

To address this problem, this project aims to develop a deep learning-based method for detecting Red Rot disease in sugarcane leaves using a large dataset of labeled images. The project will train and evaluate five different CNN models, namely VGG16, ResNet50, MobileNet, Inception V3, and a hybrid model, using the image dataset along with Adam's, SGD, and RMSProp optimizers for enhancing the model's performance. The objective is to find the best method that predicts Red Rot disease in sugarcane plants with high accuracy and efficiency. The proposed method has the potential to contribute to the development of reliable methods for early disease detection in agriculture, ultimately aiding farmers in safeguarding the yield and well-being of their crops. The project's success could lead to the development of similar deep learning-based methods for detecting other crop diseases, further advancing the field of precision agriculture.

Chapter 2

LITERATURE REVIEW

Santhrupth B.C and Devaraj Verma have focused on disease detection in sugarcane plants. They have utilized the methods such as Support Vector Machine, Random Forest and Convolutional Neural Network for the classification of the disease. The model was trained and conducted test on a dataset of 2521 high-resolution images, which enables the machine learning algorithm to use pixel to pixel in identification of the type of leaf disease. They have classified the fungal diseases into 4 classes named as red rot, rust, mosaic, and yellow leaves and a healthy class. Among all the three algorithms CNN has resulted in the highest accuracy achieving about 64% for the detection of all five classes of fungal diseases [1].

Swapnil Dadabhau Daphal and Sanjay M. Koli have investigated the performance of several deep learning algorithms described as VGG19, ResNet-50, XceptionNet, MobileNetV2, and EfficientNet B7 with transfer learning and ensemble techniques. Upon training these models with the dataset consisting of 2569 images, which were categorized into five classes of fungal diseases. They used stacking, a machine learning ensemble technique that combines many models to improve prediction accuracy, in conjunction with a transfer learning-based strategy to reach a maximum accuracy of 84% using MobileNetV2 across all five classes [2].

Visheh Tanwar, Shweta Lamba employed the convolutional neural network (CNN) to classify sugarcane leaf images into three distinct categories, aimed at identifying the leaves as healthy or unhealthy. During the process of experimentation and training of the algorithms with the dataset, they found out that the CNN model achieved an accuracy rate of 97% after 60 epochs. In the same way when they have trained the dataset with the other algorithms such as SVM and KNN models, which resulted in lower accuracy in comparison to the CNN model [3].

Shivani Machha, Nikita Jadhav has worked on disease identification across five different crops. They utilized a dataset, which included more than 300 photos of each condition, which was used to train the model. They have conducted two experiments depending on the epoch count and the images present in the dataset. In their initial experiment they have considered an epoch count of 10 and image dataset of more than 300 images, and in their second experiment they have considered epoch count of 5 and less than 100 images. Through the analysis of both experiments, they concluded that the MobileNet algorithm has achieved the highest accuracy, reaching 97.33% with an epoch count of 10. On the other hand, the identical MobileNet model's performance revealed significantly worse results than the initial findings when the epoch count, and dataset size were changed [4].

Akshara Avinash Sarode, Sufyian Salim Posharkar have worked on enhancing the sugarcane disease through transfer learning and deep convolutional networks. They have worked on training the dataset to the models like VGG16, VGG19, MobileNet, DenseNet, ResNet50, EfficientNet, AlexNet for the prediction of leaf disease by utilizing the Adam optimizer, which helps in attaining higher accuracy. They have mainly focused on testing the model on three diseases Red Rot, Red Rust and Bacterial Blight. Upon training and testing the data with all the models. Out of all the models ResNet101 has achieved the highest accuracy and performance [5].

Militante S.V, Gerardo work describes the integration of various CNN architectures to achieve excellent accuracy in detecting sugarcane illnesses using 13,725 photos. The VGGNet model outperformed the other two models, with an accuracy rate of 95.40%, next to LeNet (93.65%) as well as StridedNet (90.10%). The ability of these models to successfully classify healthy and diseased sugarcane leaves illustrates their potential for creating a system for identifying sugarcane illnesses [6].

Vaishali Wadhe, Rashmi Dongre has performed analysis on the factors which affect the crop yield of sugarcane. The analysis resulted 57% depends on the others, 23% depends on the weather conditions, 14% depends on the Pesticides and 6% depends on the insects. For the early detection of the crop being affected by the diseases they have implemented a machine learning method and proposed MobileNet model. The dataset utilized consists of four classes namely red rot, wheat rust, yellow leaf and eye spot. They considered a data split of 80:20 for training and testing purposes. They have utilized two algorithms MobileNet is also a transfer learning algorithm and CNN. They have designed a web page for disease detection. By uploading the picture, the web page displays the percentage of the disease present in the image and also if there is any presence of the disease, it will direct to the remedy page which will help the farmers to eradicate further transmission of the disease [7].

Tanwar, Swetha Lamba, Bhanu Sharma have focused on the traditional methods for the prediction of disease on the sugarcane crop. They stated the disadvantages of the traditional method like inaccuracy in the prediction, high charge for the skilled professional for the visual detection. To overcome these situations, they have proposed a Convolutional Neural Networks model for the red rot disease prediction. They considered a dataset of 490 healthy and 550 of unhealthy images which are infected or diseased leaves of the crop. They have pre-processed the model by normalizing the pictures which boost contrast and rescaled to picture size to maintain dimensions in a consistent manner. They have included convolutional layers of sequential pairs and max pooling layers to get hold of the feature. The final model is trained with epoch as 10, which resulted in achieving the accuracy rate of 93.5% [8].

Nattapak Lawanwong, Suree Pumrin has included that among the crops in Thailand, one of the key crops for the products manufacturing and economy is sugarcane. They have stated that 50% to 65% of the diseases identified are fungal diseases in sugarcane crops. For early detection of the fungal disease, they have proposed the ResNet50 model. Initially without implementing the augmentation of the data and feature extraction process the model achieved an accuracy rate of 73.20%. Later on, when the model undergoes the learning process which includes augmentation of data and extraction of the feature from the image the accuracy rate increased to 81.70% [9].

Thilagavathi, K., Kavitha, K., have stated that due to high sensitivity conditions towards the environment, the crops will be easily affected by many pests and diseases. They have worked on detecting the disease with the utilization of image processing tasks and focused on developing web applications for displaying the detection for the users. They have implemented AHE (Adaptive Histogram Equalization) for preprocessing the images, before segmenting them using the k-means clustering technique. For calculation of statistical factors like variance, mean, covariance, standard deviation they have utilized PCA (Principal Component Analysis) and GLCM (Gray Level Co-occurrence Matrix). They have worked this using the Support Vector Machine model, which resulted in an accuracy rate of 95% [10].

Lakshmikanth Paleti, Nagasri Arava have worked on identification and classification of diseases on leaves of the sugarcane. The dataset they have worked on consists of five classes like Red rot, rust, sugarcane borer wilt and healthy. They have implemented the dataset of images on KNN, SVM, ANN (Artificial Neural Network) and CNN models. After testing on all the models they have achieved an accuracy of 88% and an error rate of 12%. Also they stated that on the web application they have developed which is helpful for the detection of the disease on the leaves, Artificial Neural Network can easily detect the damages on the leaves of the crop [11].

Chapter 3

METHODOLOGY

3.1 DATA DESCRIPTION

The dataset was imported from Kaggle which consists of 862 images of the leaves of the sugarcane. The images are classified into two folders- one with 427 images which consists of healthy leaf images and the other folder consists of 435 images of the leaves which are infected by the red rot disease. Red rot disease is a fungal infection which possess a significant threat to the sugarcane crops globally. Before training the dataset with the machine learning models, initially we need to load the dataset and preprocess the images. On completing this process, the data gets formatted correctly and we can perform analysis. This is the base for training the models which helps in identification and classification of the sugarcane leaves as healthy and unhealthy accurately.

Figure 3.1: Healthy Sugarcane Leaf

Figure 3.2: Unhealthy Sugarcane Leaf

3.1.1 Data Importing

During the process of preparation of the dataset for training the model, the images are classified as healthy and unhealthy which are leaves infected with red rot disease was organized into separate folders. By taking the advantage of the capabilities the Keras preprocessing module has, the load_img function plays a vital role in efficiently loading the images from the folders. Keras offers user-friendly interface to build and train the neural networks. Keras module integrates perfectly with other deep learning libraries and framework. This provides a unified ecosystem for the entire deep learning workflow.

During the process of loading the images, a major step which involves resizing of the images into a uniform dimension of pixel size 224x224. This ensures standard consistency across the dataset. Later the load_img function transformed the loaded images into Python Imaging Library (PIL) format. It is a flexible image processing library. This process made it easier to perform other preprocessing operations and simplified the entire preparation pipeline of data for subsequent analysis and model training.

3.1.2 Data Preprocessing

Before training the dataset, the loaded images undergo preprocessing which helps to enhance the model performance. The preprocessing process bring in the normalization and the process of image data to a format fit for the neural network models. The preprocess_input function from the respective of the model applications - Mobilenet, InceptionV3, VGG16, ResNet50 was applied for each image. This function applies the relevant preprocessing steps like- mean, scaling based depending on the model's requirements. In this project for example the model InceptionV3, the function preprocess_input has performed scaling of the pixel values in range [-1,1] to match the input distribution of the model.

3.1.3 Data Organisation

After completion of the loading and preprocessing steps, the image data and the corresponding labels are organized into numpy arrays. The image is represented as a multi-dimensional array of pixel values. The labels were encoded as binary values, 1 for the healthy sugarcane leaf and 0 for the sugarcane leaf which is affected by the red rot disease. These data arrays are served as an input for training the model and the evaluation of the model.

3.2 CONVOLUTIONAL NEURAL NETWORKS (CNN'S)

Convolutional Neural Networks (CNNs) exhibit neural network architecture which are well defined for the image processing and signal processing tasks. CNNs have achieved highest success rate in analysing and identifying patterns and helps in the feature extraction process from the data. This nature helps for us in making this architecture as a great choice for the computer vision related applications, more likely image classification, object detection.

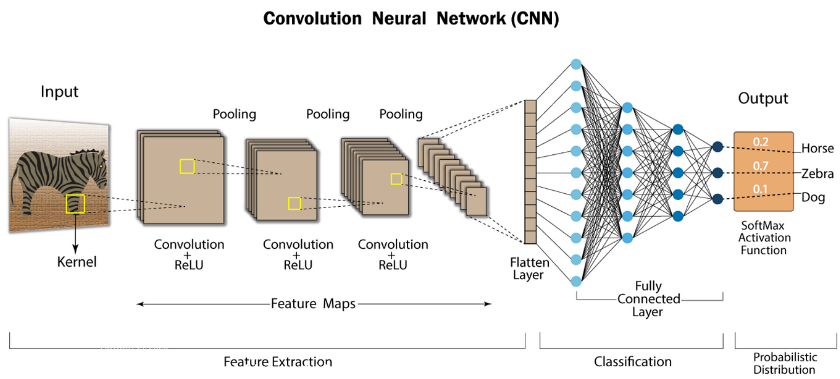

Figure 3.3: CNN Architecture [12]

CNN architecture comprised of layers for performing certain operations like convolutions, pooling, activation and fully connected operations(layers). The following information describes regarding them:

1. Convolutional Layers: These layers are helpful for extraction of the features in the given data. A feature map is produced by Convolutional layers, by applying a set of filters on the input data. During the process each filter is convolved across the input data, resulting in a new feature map that emphasizes the existence of the features that the filter is designed to detect. In Convolutional layers, the lower layers help in the identification of simple features for example edges and corners and the higher layers helps in the detections of the more complex features like, shapes and objects.

2. Pooling Layers: These layers are also known as Down sampling layers. These pooling layers are responsible for the diminishing of spatial dimensions feature maps which is produced by convolutional layers. This process helps in the prevention of over-fitting and also reduces the computational complexity of the model. These layers function by down sampling the feature map, by considering the average or the maximum value within the sliding window.

3. Activation Functions: These functions are applied after convolutional layers and fully connected layers element wise. Activation layers introduce non-linearity into the model, which allows the model understand further complex relations between the feature and output. Some of the activation's functions include Sigmoid, Tanh (Tangent Hyperbolic), ReLU (Rectified Linear Unit), Softmax function.

4. Fully Connected Layers: These layers perform the final classification task, where it connects the information from the initial layers - convolutional layers, pooling layers to the output layer. These layers exhibit the structure where each neuron is connected all the neurons present in the previous layer which is similar to the traditional neural network layers.

Upon combining all the factors mentioned above, The CNN can automatically learn and extract the features from the images. In this project, CNN is helpful for the classification of images of sugarcane leaves as two types, healthy and leaves infected with red rot disease (unhealthy). In this classification process, Convolutional layers can learn and help in the detection of the features that related to red rot disease for example leaf marks or leaf discoloration. Pooling layers help in the reduction of feature maps spatial dimensions which decreases the computational complexity of the model. Activation function helps the model to understand the complex relationships between the disease and the feature by introducing the non-linearity into the model. Upon performing all, the final classification task is carried out by the fully connected layers, which assigns a probability score to each class.

3.3 TRANSFER LEARNING

In the Deep learning techniques, one of the techniques is Transfer learning. Transfer learning is designed for learning the features which can be applied to various signal and image processing tasks. For instance, if we considered a pre-trained model which has been trained on large dataset of natural photographs, for example ImageNet. It can be tuned for another task like medical image analysis, by retraining the final layers of the network only.

The early convolutional layers of the pre-trained CNN, which is utilized for transfer learning, understands and learn the low-level features like corners, edges, textures. These features are applicable for a wide range of image processing jobs.

For identifying the red rot disease, we can utilize a pre-trained CNN model which has been trained on large dataset of natural photographs as the initial process. The initial layers of the model, learns the low-level features, which can be utilized for another task for prediction of red rot disease. Later the layers of the pre-trained CNN, learns the features at higher level comparatively specific to the original task. It can be modulated, or it can be replaced with new layers for performing the disease prediction.

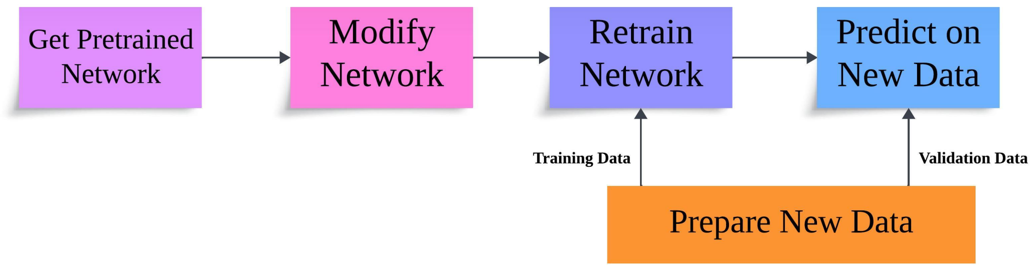

Figure 3.4: Transfer Learning

Transfer learning in addition to CNN will save time and computational resources, because we don't have the need to train the early layers from scratch. Also, as the early layers have already trained on the features that are related to the new task, this can improve the performance of the model.

3.4 CNN ARCHITECTURES

In this project, we will be utilizing many CNN architectures which includes MobileNet, InceptionV3, VGG16 and ResNet50. These algorithms are used mostly when there is work related to the computer vision field and resulted in the great performance in tasks related to image processing.

3.4.1 Mobile Net

It is a type of convolutional neural network (CNN) architecture created by Google researchers specifically for embedded and mobile vision applications. It focuses on retaining high accuracy in tasks like object identification and image classification while being lightweight and computationally economical. Important characteristics include the ability to adjust the size, speed, and accuracy of the model using hyper parameters like width multiplier and resolution multiplier, as well as depth wise separable convolutions, which lower parameters and calculations.

In this project, MobileNet can be utilized as a pre-trained model for transfer learning. By fine tuning the model with our dataset we can utilize the model for the disease prediction. For the prediction of the disease, it involves replacement of the final layers of the model which is pre-trained with new layers which help in the detection of the red rot disease, and later train the model with the dataset. During the training process, the weights of the pre-trained layers will be frozen, while the weights of the new layers will be updated which helps the model to learn the main features of the disease.

There are many advantages by implementing MobileNet in the disease prediction. It is a lightweight model so it can be run on devices which are limited on resources like smart phones, embedded systems. This will be well suitable for the farmers who can use this in their fields which don't require high computational resources. MobileNet resulted in achieving highest accuracy on different image classifications, which makes it a good choice to utilize this in the disease prediction. Since MobileNet is a pretrained model, we can use it as initial step for transfer learning, which allows to hold the knowledge which the model has gained.

3.4.2 InceptionV3

Google created the InceptionV3 convolutional neural network architecture, which uses differently sized convolutional filters to extract features on multiple scales. Inception modules, which are made up of 1x1, 3x3, and 5x5 convolutions, are included in its architecture to effectively capture both local and global data. Factorization methods lower computing expenses, while more classifiers improve training stability. It reaches quick convergence using ReLU activation functions and batch normalization. InceptionV3, which is commonly pre trained on ImageNet, is a popular tool for transfer learning in computer vision applications, providing a fair trade-off between efficiency and performance for tasks like object identification and image categorization.

In this project, InceptionV3 model is utilized for extracting the features from the leaves images and classify them as either healthy or unhealthy (red rot infected leaves). For this process, we need to preprocess the images by resizing the images to a fixed size and normalize the pixel values. Then we will input the pre-processed images into the model and train the model using the labelled dataset of leaf images of sugarcane.

During the training process, the model will adjust its weights and minimizes the loss function to improve the ability for the classification of the images. On training the model, we can make use of it for the prediction of new, unseen sugarcane leaf images. This will be helpful for the farmers to detect the disease and take preventive measures for the disease in the early stage. This can help in the reduction of loss of yields and improves the health of the crop.

3.4.3 VGG16

It is a convolutional neural network architecture developed by the Visual Geometry Group (VGG) at the University of Oxford. It was first shown in 2014 and has 16 layers total—13 convolutional layers and 3 fully connected layers. VGG-16 can extract complex features from input images by using stacked convolutions and narrow receptive fields (3x3 filters). ReLU activation functions add a non-linear component, while max-pooling layers facilitate the extraction of features. The depth of the model allows it to efficiently learn hierarchical representations. VGG-16 is frequently used for transfer learning in a variety of computer vision tasks, having been pretrained on sizable datasets such as ImageNet. It is frequently used in research and real-world applications, and despite its simplicity, it provides a solid baseline.

This model is a powerful tool for image classification works. It is much effective in extracting features and patterns present in the images. But the model is relatively large and computationally expensive. This makes the situation difficult for the deployment of the model on the resource limited devices. To handle these situations, we have also included the mobilenet and inceptionv3 models.

3.4.4 ResNet50

Microsoft Research proposed a novel convolutional neural network (CNN) architecture known as ResNet, or Residual Network. Skip connections, which enable it to understand residual functions and efficiently train very deep networks, are its primary breakthrough. ResNet uses residual blocks with skip connections to prevent the vanishing gradient problem and allow gradients to propagate. Bottleneck blocks are incorporated into the architecture to lower computing complexity in more advanced forms, such as ResNet-50 and higher. Because pre-trained ResNet models are so publicly accessible, transfer learning for tasks like object detection and picture categorization is made easier.

ResNet architectures are essential for developing computer vision research and applications because they achieve cutting-edge performance while being computationally efficient. ResNet might require more training data and computational resources when compared to other architectures.

3.5 OPTIMIZERS

Optimizers are methods or algorithms which are used to update the weights of a neural network during the training. The main aim of an optimizer is to reduce the loss function of the network and will measure the performance of the network on a given task. There are different optimizers available which have their own strengths and weakness. We have implemented three optimizers for this project – Adam optimizer, SGD optimizer, RMSProp optimizer.

3.5.1 Adam Optimizer

ADAM (Adaptive Moment Estimation) optimizer has the other two optimizers SGD and RMSProp inbuilt. It uses both momentum and adaptive adjustments to alter the learning rates for each parameter based on the average of previous gradients. ADAM optimizer uses adaptive learning rate which is based on first moment (which is also called the mean), which is an estimate of expected value of the gradient and second moment (also called as the uncentered variance), which is an estimate of the expected value of squared gradient. To address early biases in gradient estimations, bias correction is incorporated. Adam is robust across a range of applications and architectures and requires very little hyperparameter adjustment. Because of its adaptability and parallelizability, deep neural network training with it is rather popular.

The size of steps taken by the optimizer is determined by the learning rate, which is multiplied by the first moment. If the learning rate is high, it can lead the optimizer to overshoot the minimum of the loss function. If the learning rate is low, it leads the optimizer to converge slowly.ADAM optimizer consists of many hyperparameters which can be tuned with respect to the problem and the dataset. The important hyperparameters are – learning rate, beta1, beta2 and epsilon parameter.

The parameters which control the running averages of the first and second moments are beta1 and beta2. The parameter beta1 regulates the weight specified to the previous first moment and the parameter beta2 determines the weights specified to the previous second moment.

$$V_t = \beta_1 * V_{t-1} - (1-\beta_1) * g_t \tag{3.1}$$

$$S_t = \beta_2 * S_{t-1} - (1-\beta_2) * g_t^2 \tag{3.2}$$

$$\triangle\omega_t = -\eta \frac{\nu_t}{\sqrt{S_t + \epsilon}} * g_t \tag{3.3}$$

Where, η : *Initial Learning rate*

g_t : *Gradient at time t along* ω_j

V_t : *Exponential Average of gradients along* ω_j

S_t : *Exponential Average of squares of gradients along* ω_j

β_1, β_2 : *Hyperparameters*

3.5.2 SGD Optimizer

In machine learning and deep learning, Stochastic Gradient Descent, or SGD, is a fundamental optimization approach. By adjusting the model parameters in accordance with the gradient of the loss function regarding the parameters, iteratively minimizes the loss function. Compared to traditional gradient descent, SGD is faster and more scalable since it works with mini batches of data. To adjust the step size in parameter space, a learning rate must be defined. Additionally, momentum is frequently included to speed up convergence.

SGD optimizer works by computing the gradient of the loss function in perspective of weights and biases of the network. Later the weights and biases are updated by taking a step in the direction of negative gradient.

The SDG optimizer have several variations which helps in the improvement of the performance. One such of the variation is the momentum. For the current update it adds a fraction of the previous update. This helps to smoothen the updates and improves the convergence. Another such variation is Nesterov momentum. It computes the gradient at a point that is ahead of the current point in the direction of the momentum. This will improve the convergence and avoids getting stuck in local minima. The weights assigned to the previous update is given by the momentum parameter. The momentum which has high value can help in improvement of convergence and ease the updates. Equation for updating the weights and bias in SGD.

$$w_t = w_{t-1} - \eta \frac{\delta L}{\delta w_{t-1}} \tag{3.4}$$

$$b_t = b_{t-1} - \eta \frac{\delta L}{\delta b_{t-1}} \tag{3.5}$$

Equation for updating the weights and bias in SGD with momentum.

$$w_t = w_{t-1} - \eta V_{dw_t} \tag{3.6}$$

$$b_t = b_{t-1} - \eta V_{db_t} \tag{3.7}$$

Where, $V_{dw_t} = \beta V_{dw_{t-1}} + (1-\beta)\frac{\delta L}{\delta w_{t-1}}$

$V_{db_t} = \beta V_{db_{t-1}} + (1-\beta)\frac{\delta L}{\delta w_{t-1}}$

3.5.3 RMSProp Optimizer

RMSprop (Root Mean Square Propagation) is an optimization algorithm widely used in deep learning. By modifying the learning rates for every parameter in accordance with the gradient history, it overcomes the drawbacks of stochastic gradient descent (SGD). RMSprop may automatically modify learning rates by calculating an exponentially decreasing average of the squared gradients. To stabilize updates, gradients are normalized, and a smoothing constant is utilized to regulate the pace of decay. RMSprop performs well for deep neural network training on big datasets, is computationally efficient, and requires little tweaking. All things considered, it's a strong and popular deep learning optimization approach.

RMSProp optimizer dynamically adjusts the learning rate, which is based on the historical gradient information. It allows the model to converge faster and more accurate. This is commonly useful for the deep learning applications like non-stationary or sparse gradients. By the dynamic adjustment of the learning rate, it helps to avoid the problem of the learning rate value being too high or too low, which leads to slow convergence or overfitting.

$$E[g^2]_t = \beta E[g^2]_{t-1} + (1-\beta)(\frac{\delta C}{\delta W})^2 \tag{3.8}$$

$$W_t = W_{t-1} - \frac{n}{\sqrt{E[g^2]_t}}\frac{\delta C}{\delta W} \tag{3.9}$$

Where, $E[g]$: *Moving average of squared gradients.*

$\frac{\delta C}{\delta W}$: *gradient of the cost function with respect to the weight.*

n : *Learning data.*

β : *moving average parameter(default value - 0.9)*

Chapter 4

MODEL EVALUATION

4.1 MOBILENET MODEL

MobileNet	ADAM	SGD	RMSProp
(80-20)	0.972	0.971	0.970

Table 4.1: Accuracies of MobileNet Model

For MobileNet, the process of evaluation on MobileNet model includes training the model with the three optimizers - ADAM, SGD, RMSProp. Upon considering the dataset with split ratio of 80:20 delivered useful insights into their performance.

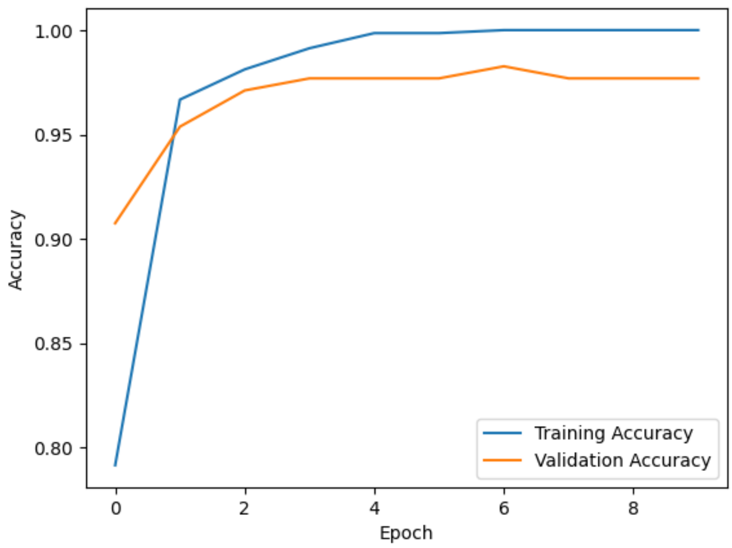

Figure 4.1: MobileNet(ADAM) Training and Validation Accuracy

Upon training the dataset with different models with different optimizers, the MobileNet model trained with Adam optimizer achieved the maximum accuracy rate of 0.972 , which showcases the robustness in MobileNet architecture optimization for the considered dataset. The training history of the MobileNet model along with ADAM optimizer enhances the models effective learning proficiency.

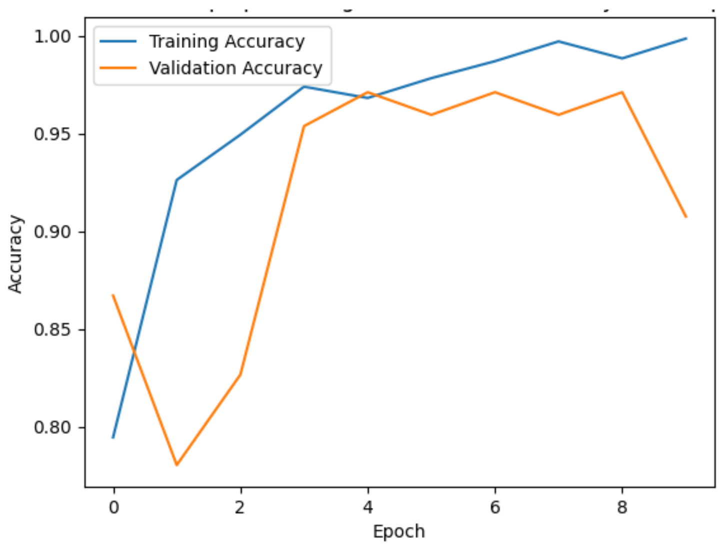

Figure 4.2: MobileNet(SGD) Training and Validation Accuracy

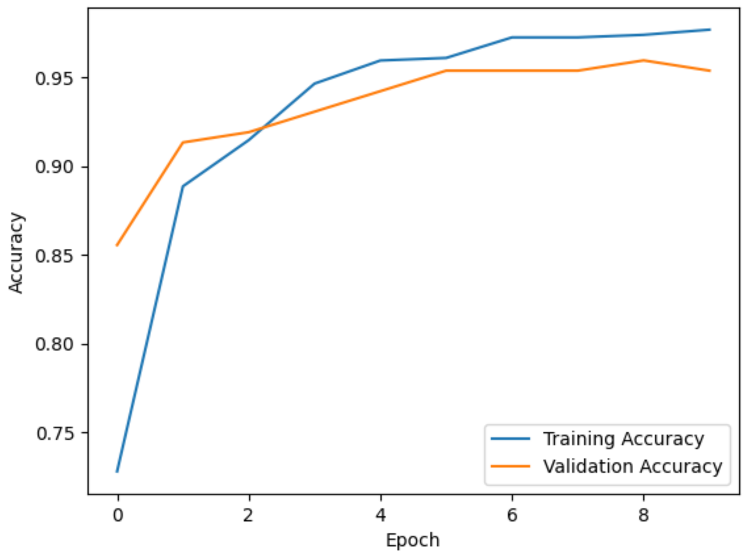

Figure 4.3: MobileNet(RMSProp) Training and Validation Accuracy

On comparing, the MobileNet model which is trained using the SGD and RMSProp optimizers exhibits a slower rate of convergence and a decrement in overall accuracy. When the model is trained using the SGD optimizer resulted with an accuracy rate of 0.971 and when the model is trained using the RMSProp optimizer the achieved accuracy rate is 0.970.

Selection of the optimizers is also important because, there would be an effect on the model performance and also the convergence behaviour will be highlighted by this analysis. Considering the factors, in this situation ADAM optimizer is recommended for the improvement of efficiency during the training process and accuracy for the image classification tasks through MobileNet model.

4.2 INCEPTIONV3 & VGG16 MODELS

InceptionV3	Adam	SGD	rmsProp
(80-20)	0.901	0.895	0.913

Table 4.2: Accuracies of InceptionV3 Model

VGG16	Adam	SGD	rmsProp
(80-20)	0.878	0.728	0.855

Table 4.3: Accuracies of VGG16 Model

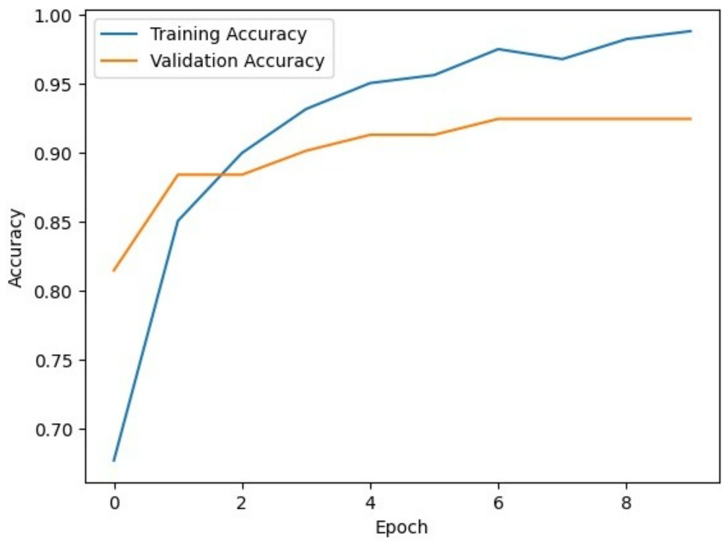

Figure 4.4: InceptionV3(ADAM) Training and Validation Accuracy

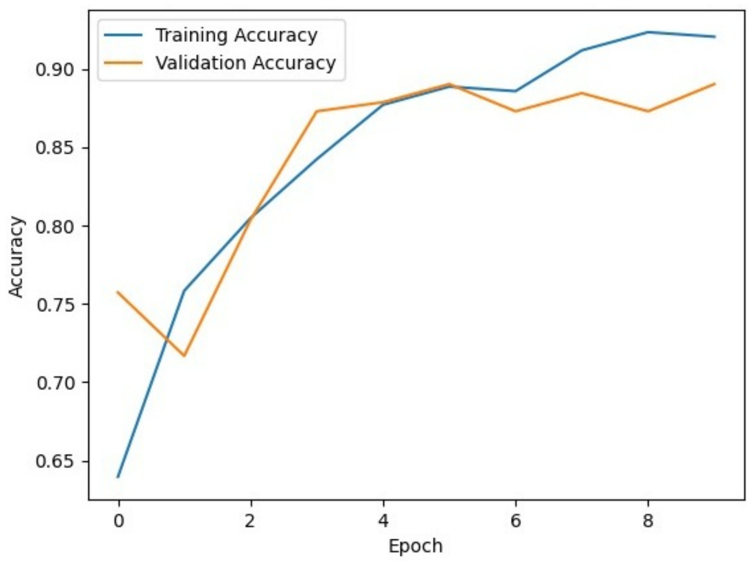

Figure 4.5: VGG16(ADAM) Training and Validation Accuracy

For InceptionV3 & VGG16, the performance of the models VGG16 and InceptionV3 when trained with the three optimizers with the data split ration of 80-20, provided some valuable insights into their performance.

The VGG16 model when trained with the ADAM optimizer, it achieved the highest accuracy of 0.878, which showcases the productivity in the optimization of the VGG16 architecture on the given dataset. Considering the previous training history of the VGG16 model along with the Adam optimizer projects a steady improvement with respect to accuracy over epoch which indicates robust learning capabilities.

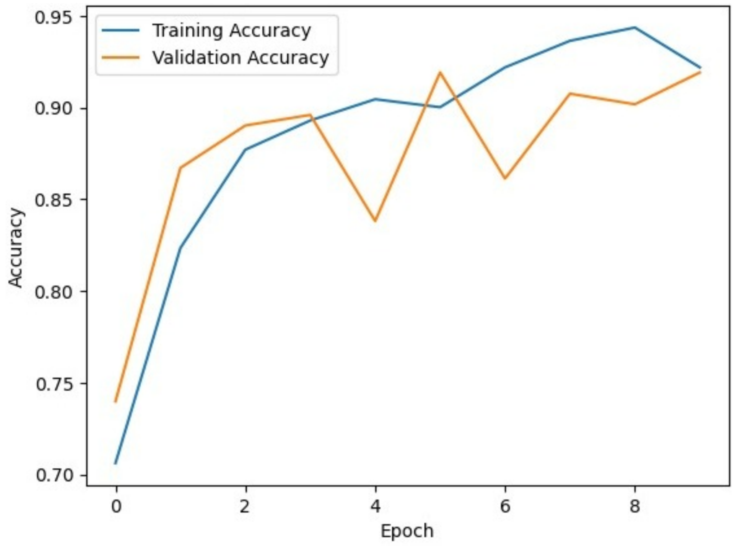

Figure 4.6: InceptionV3(SDG) Training and Validation Accuracy

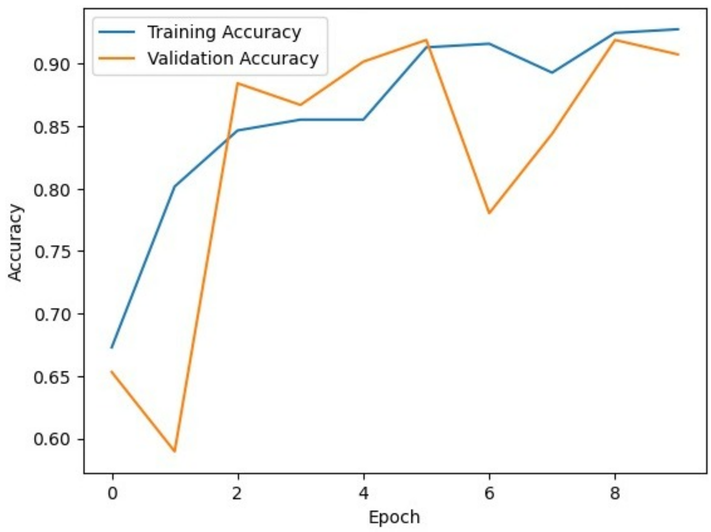

Figure 4.7: InceptionV3(RMSProp) Training and Validation Accuracy

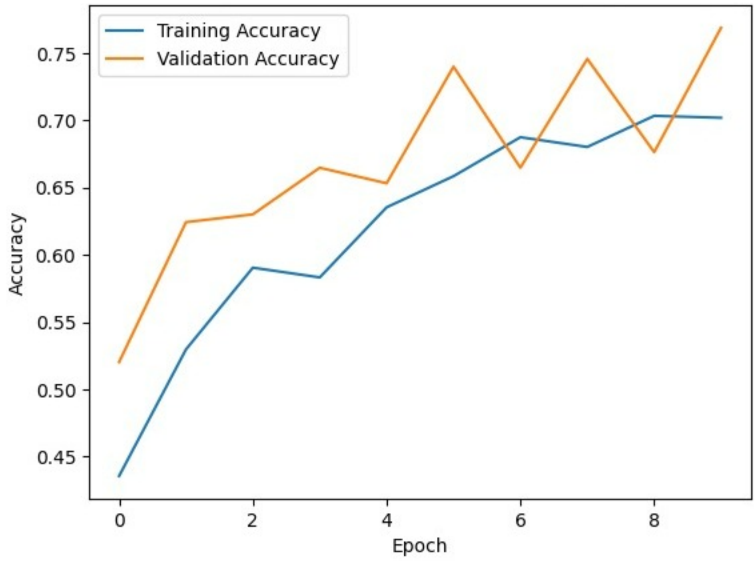

Figure 4.8: VGG16(SDG) Training and Validation Accuracy

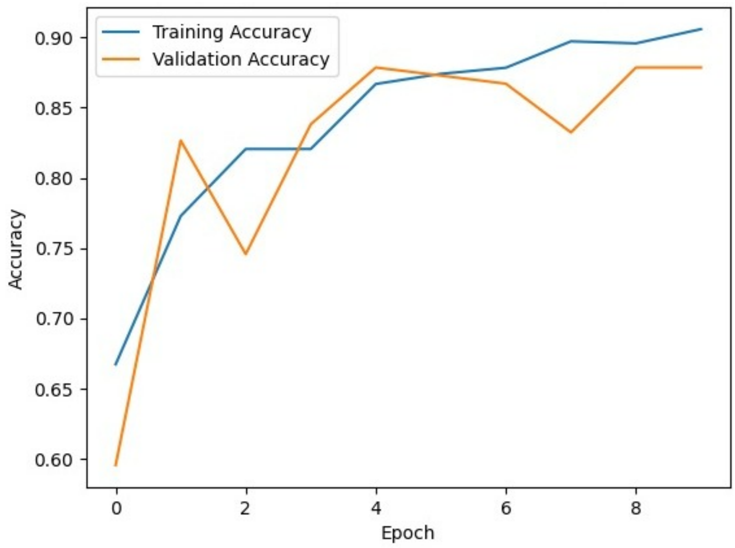

Figure 4.9: VGG16(RMSProp) Training and Validation Accuracy

Similarly, when the VGG16 model is trained with other optimizers – SGD and RMSProp, they have exhibited lower accuracy rate and a slower convergence. The model achieved an accuracy rate of 0.742, when it is trained with SGD optimizer, while the same model when trained with the RMSProp achieved an accuracy rate of 0.869. Particularly, when the model is trained with the dataset while using the RMSprop as optimizer, it projected some fluctuations in the accuracy during the epoch training. The InceptionV3 model when trained with Adam optimizer has resulted with an accuracy rate of 0.901, which indicates the effectiveness for optimization of the Inception V3 architecture for the considered dataset. In the same way, the InceptionV3 models when undergoes training with the RMSProp optimizer it also resulted well with an accuracy rate of 0.913. Somehow, the InceptionV3 model when underwent training with the SGD optimizer, we noticed that this model lags slightly with an accuracy rate of 0.903.

Hence, the decision of selecting optimizers also impacts the model's performance and the convergence behaviour for both the InceptionV3 and VGG16 architectures. After the analysis we have observed the Adam optimizer and RMSProp optimizer has tend to perform well and achieve good accuracy when compared to the SGD optimizer. In many situations, the Adam optimizer outperformed well and achieved higher accuracy among many optimizers.

4.3 RESNET50 MODEL

ResNet50	Adam	SGD	rmsProp
(80-20)	0.745	0.618	0.722

Table 4.4: Accuracies of ResNet50 Model

The ResNet50 model when trained by using the optimizers we have considered and with a data split ratio of 80:20, reveals some variation in the performance levels of the model. When the model undergoes training by using the Adam optimizer, the model has achieved an accuracy rate of 0.745.

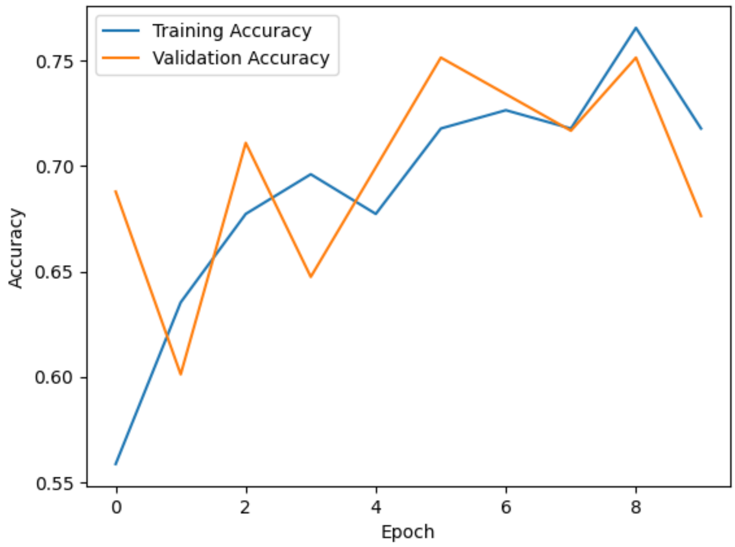

Figure 4.10: ResNet50(ADAM) Training and Validation Accuracy

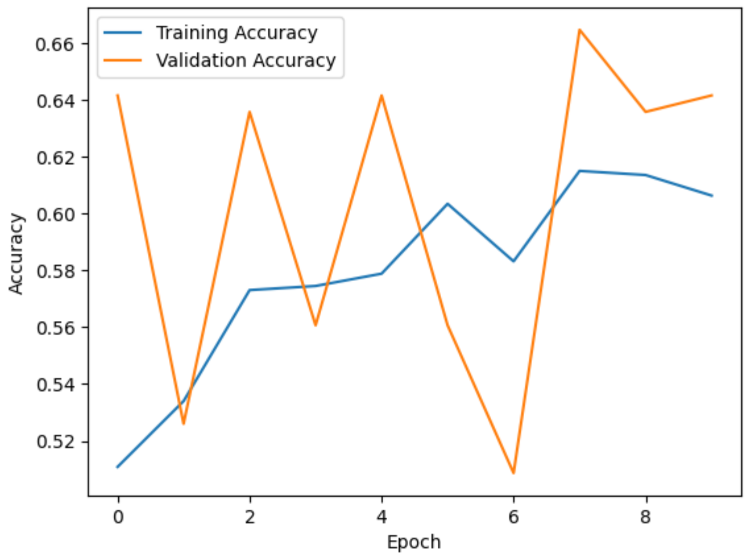

Figure 4.11: ResNet50(SGD) Training and Validation Accuracy

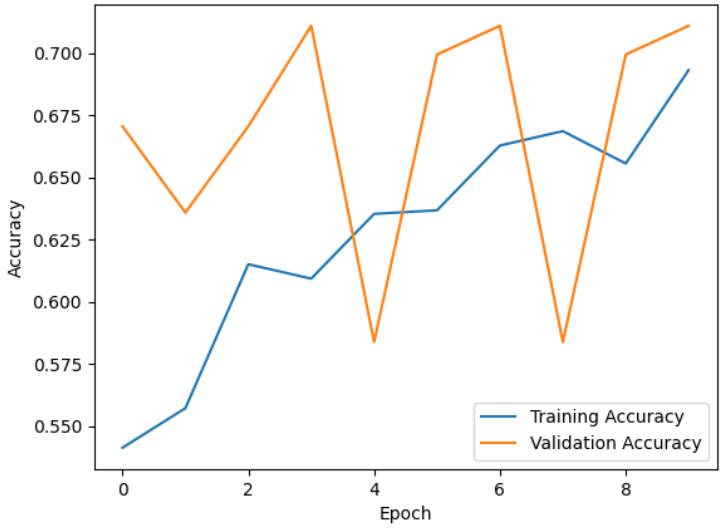

Figure 4.12: ResNet50(RMSProp) Training and Validation Accuracy

On the other hand, the training history shows varying results, which indicates convergence complications. Later when the model undergoes training by utilizing SGD optimizer, the model achieved a lower accuracy rate of 0.707. The training process with the model also displayed inconsistency and indication of difficulties in the optimization.

At last, when the model is trained using RMSprop optimizer, the model achieves an accuracy rate of 0.726. This model projects a stable convergence through training epochs. Considering all the models of ResNet50, the performance of RMSprop achieves short accuracy when compared with the Adam optimizer. The choice of the optimizers also impacts the performance of the model and the convergence behaviour. So, the Adam optimizer is considered for displaying its superiority in the model performance when compared with the SGD and RMSprop optimizer.

4.4 COMPARISON OF MODEL PERFORMANCE

After analysing and performing the evaluations considering different data splits ratio (80-20, 70-30, 60-40), it is observable that some patterns emerge regarding the model performance.

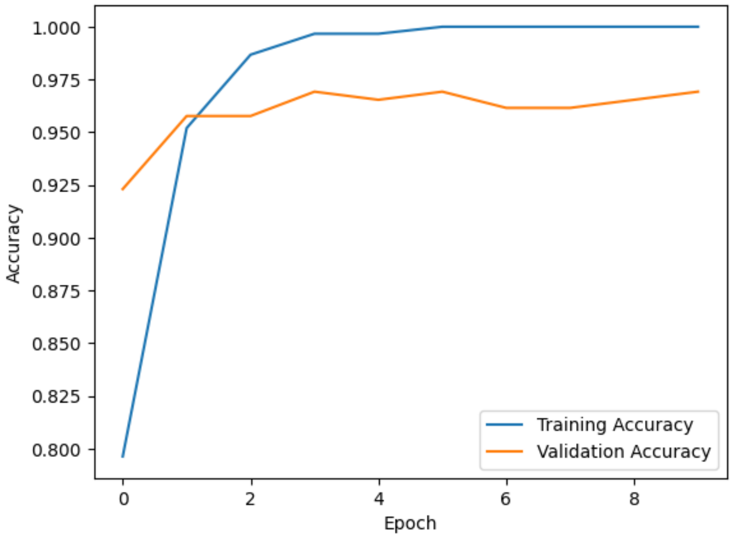

Figure 4.13: MobileNet(ADAM) Training and Validation Accuracy with (70-30) Split

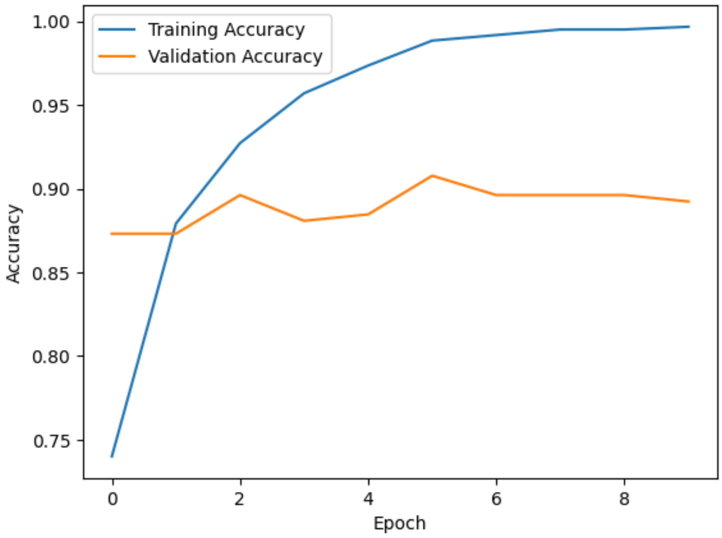

Figure 4.14: InceptionV3(ADAM) Training and Validation Accuracy with (70-30) Split

Particularly, in the dataset of 70-30 split scenario, the MobileNet model while trained using the Adam optimizer stands out as the top one, projecting an accuracy rate of 0.969. Following the model closely another model named InceptionV3 model too is behind the MobileNet model which achieved an accuracy rate of 0.903 while the model is trained with the SGD optimizer. This indicates robust performance of the models when performing with different data splits and different optimizers.

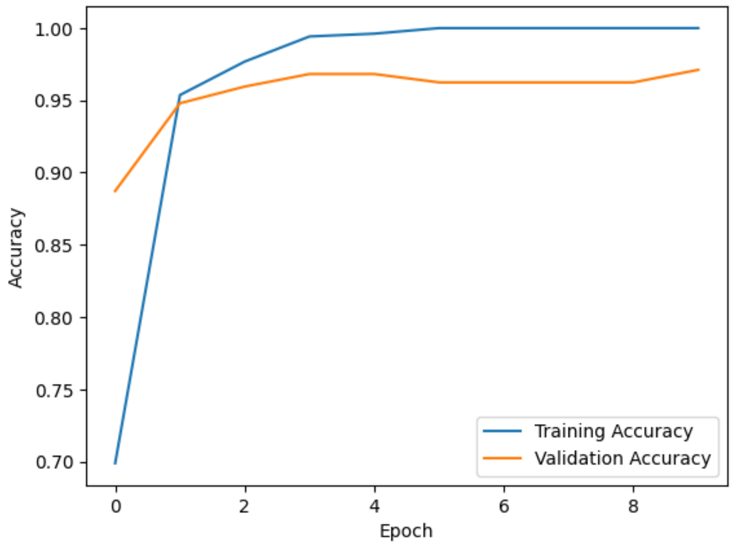

Figure 4.15: MobileNet(ADAM) Training and Validation Accuracy with (60-40) Split

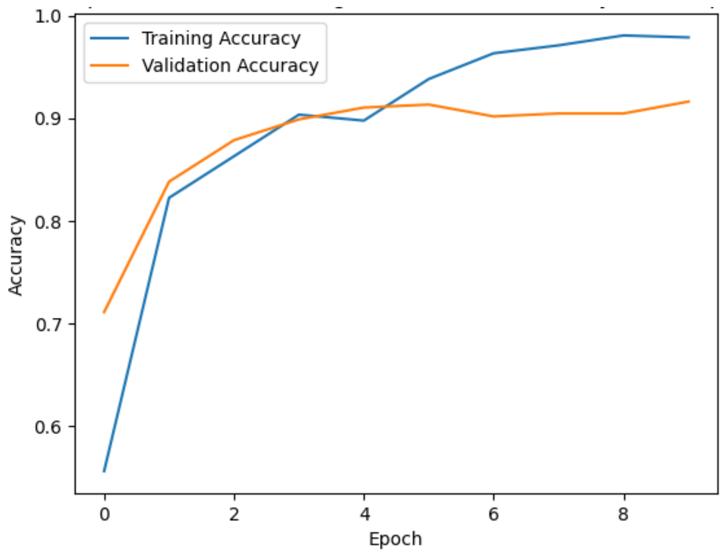

Figure 4.16: InceptionV3(ADAM) Training and Validation Accuracy with (60-40) Split

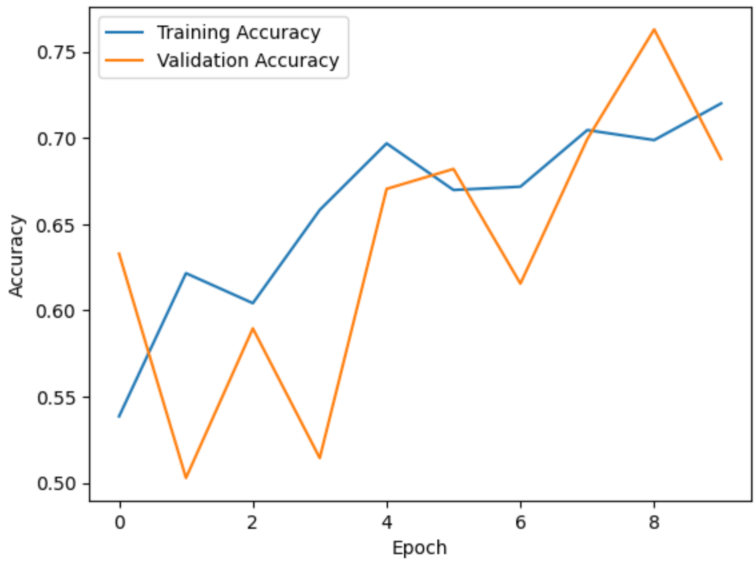

Figure 4.17: VGG16(ADAM) Training and Validation Accuracy with (60-40) Split

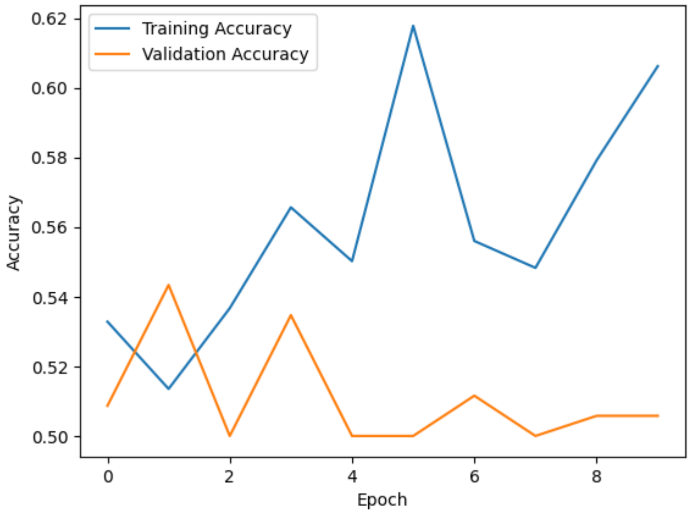

Figure 4.18: ResNet50(ADAM) Training and Validation Accuracy with (60-40) Split

While the evaluation process is performed using the dataset with split of 60-40, unexpected dynamics which are related to the model's performance emerged. It is observed that ResNet50 and VGG16 model has resulted in less efficiency when compared with the MobileNet and InceptionV3 models. Given that some models are tolerant to changes in distribution of the data than others, which raises possibility of difficulties when generalizing across various splits.

Upon considering all the evaluations for all the models, Adam optimizers exhibit high performance when compared with other optimizers. It highlights the efficiency in the optimization of model parameters for the improvement in the accuracy. Next to this, RMSProp optimizers falls in the category, which exhibits a slight lower performance next to the ADAM optimizer. This displays competitive performance compared to ADAM. These findings play crucial role in the selection of the optimizers which influence model convergence and the performance across various data distributions. This highlights the importance of the optimization strategies in deep learning model development.

4.4.1 Model Accuracy Analysis

Comparative Analysis of Optimizers for Different Pre-Trained Models. We conducted a comparative analysis that compared the performance of three different optimizers - Adam, SGD (Stochastic Gradient Descent), and RMSprop - on four pre-trained deep learning models (MobileNet, VGG16, InceptionV3, and ResNet50) for predicting sugarcane diseases. Each model was trained and tested using one of three data split ratios (80:20, 70:30, or 60:40). For all the three data split ratio, the accuracy values obtained were as follows:

Optimizers	Model - Mobile Net		
	80:20	70:30	60:40
ADAM	0.972	0.969	0.971
SGD	0.971	0.950	0.945
RMSPROP	0.970	0.965	0.953

Table 4.5: MobileNet accuracy

Optimizers	Model - InceptionV3		
	80:20	70:30	60:40
ADAM	0.901	0.900	0.913
SGD	0.895	0.903	0.884
RMSPROP	0.913	0.873	0.893

Table 4.6: InceptionV3 accuracy

Optimizers	Model - VGG16		
	80:20	70:30	60:40
ADAM	0.878	0.873	0.872
SGD	0.728	0.742	0.520
RMSPROP	0.855	0.869	0.803

Table 4.7: VGG16 accuracy

Optimizers	Model - ResNet50		
	80:20	70:30	60:40
ADAM	0.745	0.730	0.705
SGD	0.618	0.707	0.552
RMSPROP	0.772	0.726	0.502

Table 4.8: ResNet50 accuracy

Chapter 5

PROPOSED MODEL

After training and testing the dataset on different models like MobileNet, InceptionV3, VGG 16, and ResNet 50 with different optimizers like Adam, SGD, RMSProp and with different splits ratio (80- 20), (70- 30), (60-40). Based on the analysis we got to know that the MobileNet and Inception V3 model with Adam Optimizer performs well on the dataset.

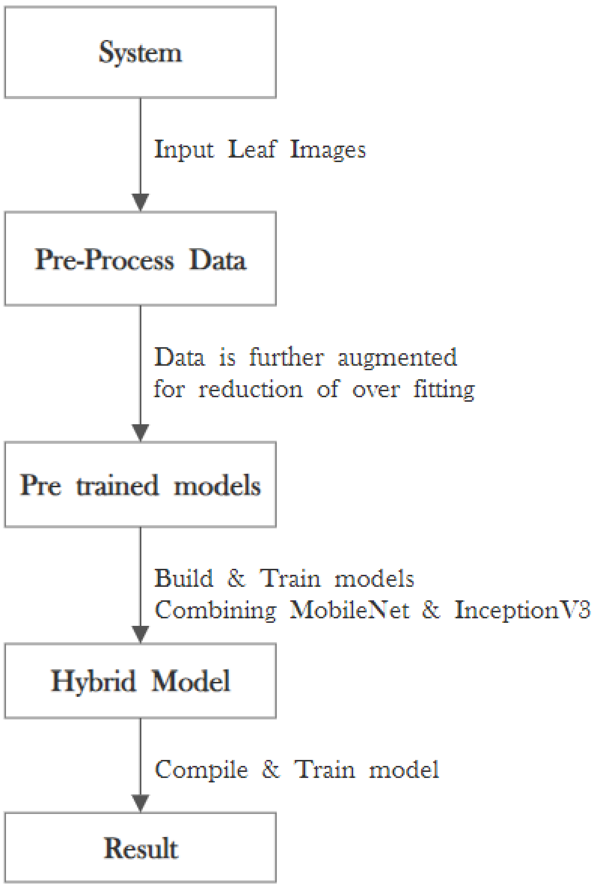

Figure 5.1: Block Diagram of the System

So, we thought there would be a change for further enhancement of accuracy of the data. So, we considered an approach to build a new Hybrid model based on the analysis. As mentioned in preprocessing steps, load each image from the dataset, transform it to an array, then preprocess it based on the needs and usefulness of our models (MobileNet or InceptionV3). Images from the ”Healthy” and ”Unhealthy” directories are processed, and the resulting images and labels are combined to form the feature matrix X and label vector Y.

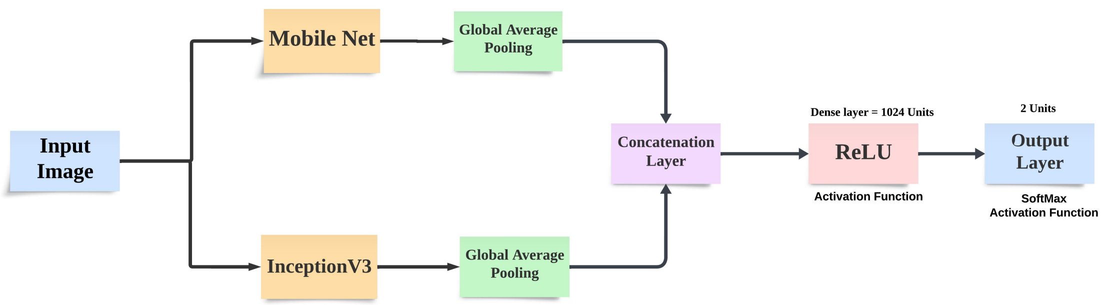

Figure 5.2: Hybrid Model

We have opted for the hybrid model architecture, to take the advantage of these existing models which performed at higher efficiency. This hybrid model architecture involves in concatenation of the global average pooling layers from both the MobileNet and InceptionV3 models. We have considered global average pooling layers because of its ability to organise the spatial information while holding on to the important features.

This helps the model to be adaptable for transfer learning, where the pre-trained models undergo fine-tuning. By integrating this layer, we aim to capture higher level features within the complete image, which allows for better feature extraction and improves the model capacity for the generalization of the unseen data.

For learning detailed and complex patterns, a dense layer with 1024 units and ReLU (Rectified Linear Unit) activation is added, followed by the output layer implementing softmax activation for binary classification. The model is compiled with the Adam optimizer and sparse categorical cross-entropy loss function, which helps in optimization of performance and convergence.

The model is trained with the training data for 10 epochs with batch size of 32, which helps the model to refine the parameters, and this improves the prediction accuracy rate. A random selection of 10 examples from the testing set is visualized along with their true and predicted labels using matplotlib. The images are displayed in a grid format with their true labels and predicted labels as titles. This helps in easy evaluation of the model's capabilities towards prediction of the disease and possible areas for improvement.

Chapter 6

RESULT

This project aimed to develop and implement a model for the prediction of red rot disease on leaves of the sugarcane crop. On performing training and testing operations we have analysed to build a hybrid model for achieving higher accuracy rate. Our previous models MobileNet and InceptionV3 achieved the highest accuracy rate of 97% and 90%, while implementing the algorithm using the Adam optimizer using a data split of 80-20. Depending on the results obtained, we decided to construct the model on combining the MobileNet model and InceptionV3 model. On considering a data split ratio of 80-20 and performed operations by training the model. The results for the hybrid model achieved an accuracy rate of 97.7%, which is slightly higher than the previous predicted models.

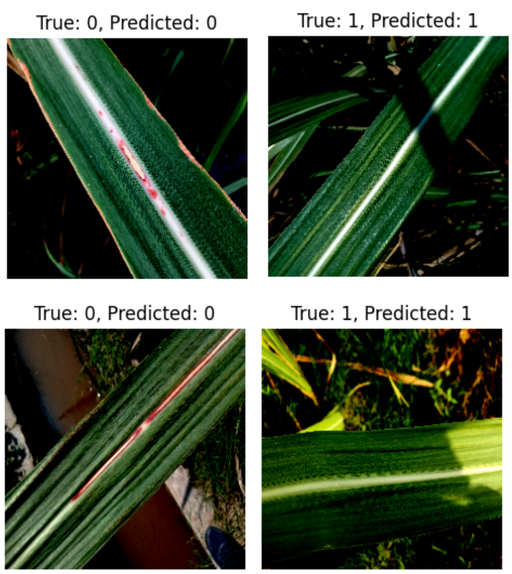

Figure 6.1: Hybrid model prediction of Red rot.

To understand and analyse more about the model we have performed confusion matrix. The confusion matrix resulted in true negative values of 41, true positives values of 44, false positive values of 2 and false negative values of 0.

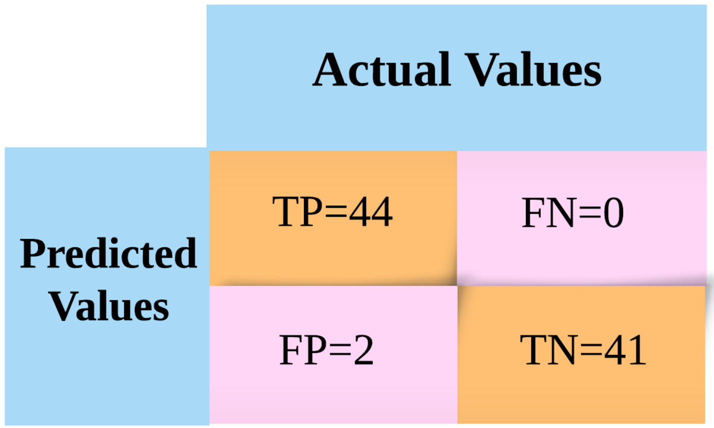

Figure 6.2: Confusion Matrix

True Positive (TP) : It describes about the situation when the actual result is positive and the model also accurately predicts the value as positive.

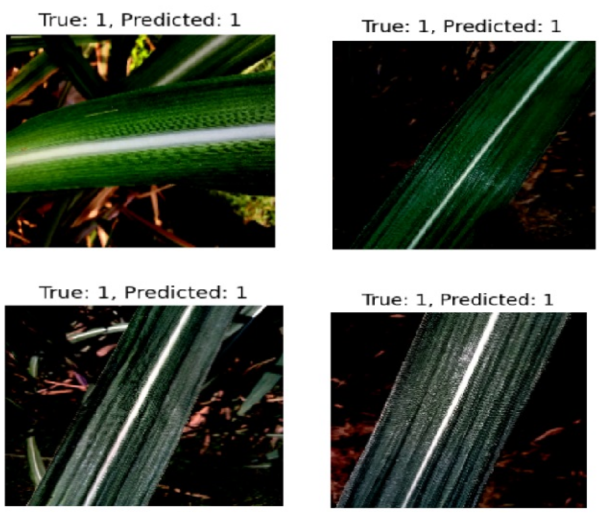

Figure 6.3: True Positive fig.

False Positive (FP): In a situation, when the actual result is negative, but the model forecasts the result as positive which results in inaccuracy.

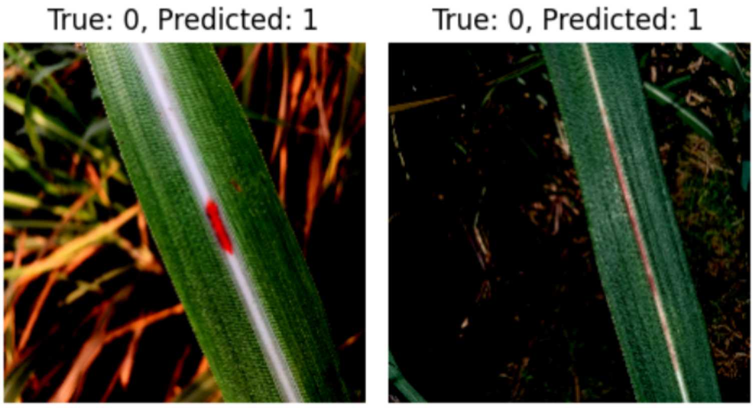

Figure 6.4: False Positive fig.

True Negative (TN): If the actual result is negative and the model also projected, the value as negative.

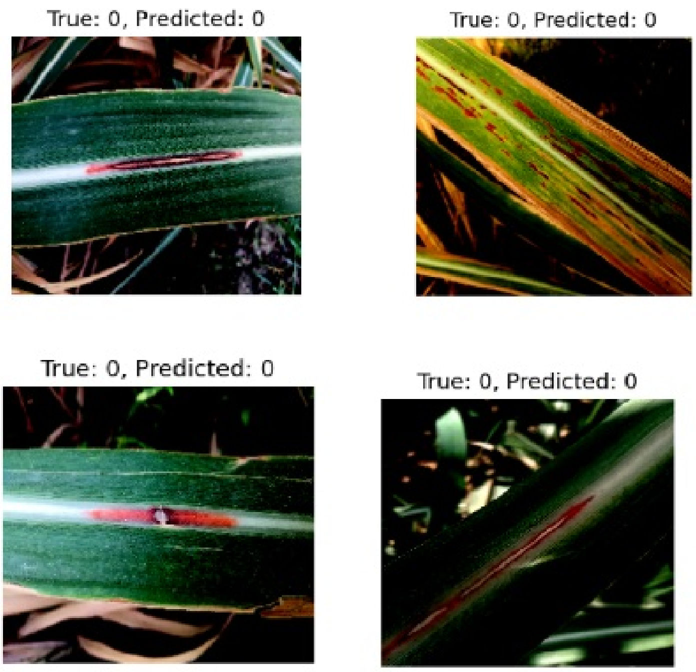

Figure 6.5: True Negative fig.

False Negative (FN): When the actual result is positive, but the model with inaccuracy returns the value as negative.

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \tag{6.1}$$

$$Precision = \frac{TP}{TP+FP} \tag{6.2}$$

$$Recall = \frac{TP}{TP+FN} \tag{6.3}$$

$$F1\_Score = \frac{2.Precision.Recall}{Precision+Recall} \tag{6.4}$$

Accuracy	0.977
Precision	0.956
Recall	1.0
F1 Score	0.977

Table 6.1: Confusion Matrix Metrics

Upon calculating the model have achieved the precision value of 0.956. This describes that if the model predicts a situation where there is a sign of positive red rot disease on the leaves, there would be high probability of the prediction is True. This is an important factor, because it makes sure that the model doesn't produce many false positives, which leads to false prediction and unwanted treatments for the crop and wastage of the finance.

On analysis, the recall value is achieved to 1. This shows that the model is able to identify all the positive cases of the red rot disease in the dataset. This helps the model no to miss any scenarios of the disease, which may lead to severe consequences.

The F1-score has resulted of value 0.978. This describes the balance between the precision and recall. The best possible value for the F1-score is 1, which indicates it is the perfect precision and recall. This elevated score achieved by the hybrid model presents that the model is reliable and accurate for the prediction of the red rot disease.

Chapter 7

CONCLUSION

With the results of our study, it is observable that the hybrid model, which is a combination of MobileNet and Inception V3 architectures, has demonstrated slightly superior accuracy compared to the other four models evaluated. The hybrid model successfully achieved its objective by accurately classifying sugarcane leaf images as healthy or unhealthy with a precision of 97.7%.

This research emphasizes the effectiveness of deep learning-based models in accurately classifying sugarcane leaf diseases, showcasing the potential for employing computer vision and machine learning techniques to aid farmers in the early detection of sugarcane diseases. This will help in allowing the farmers for taking precaution measures for the crop to avoid transfer of the disease. This will help is minimization of loss of crop and maximizing the yield.

Through the results obtained, our project might have considerable effects for the agriculture field and the prediction of the disease. Upon developing an accurate and reliable method for the prediction of the red rot disease, will help the farmers and workers in agricultural fields protecting their crop from the disease.

7.1 SCOPE OF FUTURE WORK

There is a scope to develop a user-friendly web application which can be accessible through the smart phone devices that integrates the model, which enables the detection of the disease in the fields in real time. But some of the situations like lower level of internet signal, will might affect the process while uploading the picture into the web application. So, invention of low-tech model will might be helpful for the farmers. In addition to this, we can explore some of the –+associated products costs which helps for the eradication of the diseases.

One of the factors which contributes for achieving higher accuracy rate is the images. It is important to evaluate the image datasets, which plays an important role for achieving higher accuracy. Some of the factors such as bad weathers, lightnings and the angle of the camera would affect the quality of the image. Collaborating with the farmers and acquiring good quality of images for the dataset will help the model to train at higher efficiency and achieving the results with best accuracy rate.

We can integrate the model in the drone processing unit. The drone will be equipped with GPS, which will be navigated to the sugarcane fields, captures the high-resolution images of the fields. After capturing, the integrated model will analyse the captured images, identifying the chances for the red rot infection based on the trained image features. We can also be able to develop an application for transferring all the data and analysis made by the integrated model present in the drone to the mobile. Through that alerts can be generated for the areas that have been affected by this disease.

REFERENCES

[1] **Santhrupth, B.C., 2024**, Intelligent Disease Detection in Sugarcane Plants: A Comparative Analysis of Machine Learning Models for Classification and Diagnosis. *International Journal of Intelligent Systems and Applications in Engineering*, 12(8s), pp.299-306.

[2] **Daphal, S.D. and Koli**, Enhancing sugarcane disease classification with ensemble deep learning: A comparative study with transfer learning techniques. Heliyon, 9(8).

[3] **Tanwar. V, Lamba.S**, Deep learning-based approach for leaf disease of sugarcane classification. In 2023 *IEEE 12th International Conference on Communication Systems and Network Technologies (CSNT)* (pp. 176-180). IEEE.

[4] **Machha, S., Jadhav**. Crop leaf disease diagnosis using convolutional neural network. *International Research Journal of Engineering and Technology (IRJET)*, 7(02).

[5] **Sarode, A.A.**, Posharkar.Enhanced Sugarcane Disease Diagnosis through Transfer Learning & Deep Convolutional Networks.

[6] **Militante, S.V. and Gerardo, B.D.,** 2019, December. Detecting sugarcane diseases through adaptive deep learning models of convolutional neural network. In 2019 *IEEE 6th International Conference on Engineering Technologies and Applied Sciences (ICETAS)* (pp. 1-5). IEEE.

[7] **Wadhe Vaishali, Dongre, R., Kankriya, Y. and Kuckian, A.**, 2022, December. Sugarcane Disease Detection using Deep Learning. In 2022 5th *International Conference on Advances in Science and Technology (ICAST)* (pp. 40-44). IEEE.

[8] **Tanwar, V., Lamba, S., Sharma, B. and Sharma, A.**, 2023, March. Red rot disease prediction in sugarcane using the deep learning approach. In 2023 *2nd International Conference for Innovation in Technology (INOCON)* (pp. 1-5). IEEE.

[9] **Lawanwong, N. and Pumrin, S.**, 2022, January. Development of an Algorithm for Classifying Common Sugarcane Diseases in Thailand. In The 14th Regional Conference on Electrical and Electronics Engineering (RC-EEE 2021) (p. 152).

[10] **Thilagavathi, K., Kavitha, K., Praba, R.D., Arina, S.V. and Sahana, R.C.**, 2020. Detection of diseases in sugarcane using image processing techniques. Bioscience Biotechnology Research Communications, Special Issue, (11), pp.109-115.

[11] **PALETI, L., NAGASRI, A., SUNITHA, P., SANDYA, V., SUMALLIKA, T., KANDUKURI, P. and KUMAR, K.K., 2023**. SUGAR CANE LEAF DISEASE CLASSIFICATION AND IDENTIFICATION USING DEEP MACHINE LEARNING ALGORITHMS. Journal of Theoretical and Applied Information Technology, 101(20).

[12] **N.Shahriar**, "What is Convolutional Neural Network - CNN (Deep Learning)," *Medium*, Feb. 01, 2023. https://nafizshahriar.medium.com/what-is-convolutional-neural-network-cnn-deep-learning-b3921bdd82d5.