A

Project Report

on

Plant Disease Detection Using Machine Learning and Image Segmentation Techniques

Submitted in partial fulfilment of the

requirement for the award of the

Degree of

BACHELOR OF TECHNOLOGY

in

ELECTRONICS AND COMMUNICATION ENGINEERING

by

Abhay kumar shah (19021030175)
Ansika Sinha (20151030096)

Under the Guidance of

Prof. Prabhat Kumar Srivastava
(Assistant Professor, DEECE)

GALGOTIAS
UNIVERSITY

DEPARTMENT OF ELECTRICAL, ELECTRONICS AND COMMUNICATION ENGINEERING

May, 2023

DECLARATION

We declare that the work presented in this report titled "**Plant Disease Detection Using Machine Learning and Image Segmentation Techniques**", submitted to the Department of Electrical, Electronics and Communication Engineering, Galgotias University, Greater Noida, for the Bachelor of Technology in Electronics and Communication Engineering is our original work. We have not plagiarized unless cited or the same report has not submitted anywhere for the award of any other degree. We understand that any violation of the above will be cause for disciplinary action by the university against us as per the University rule.

Place:

Date:

Signature of the Student

Abhay kumar shah (19SECE1030030)

Ansika Sinha (20SECE1030008)

Department of Electrical, Electronics and Communication Engineering

CERTIFICATE

This is to certify that the project titled **"Plant Disease Detection Using Machine Learning and Image Segmentation Techniques''** is a bonafide work carried out by **Abhay Kumar Shah and Ansika Sinha,** during the academic year 2022-23. We approve this project for submission in partial fulfilment of the requirements for the award of the Degree of Bachelor of Technology in Electronics and Communication Engineering, Galgotias University.

Project Guide: **Prof. Prabhat Kumar Srivastava**

Internal Examiner

External Examiner

Approved by

Head

ACKNOWLEDGEMENT

We are grateful to The Department of Electrical, Electronics and Communication Engineering for giving us the opportunity to carry out this project, which is an integral fragment of the curriculum in Bachelor of Technology program at the Galgotias University, Greater Noida. We would like to express our heartfelt gratitude and regards to our project guide, **Prof. Prabhat Kumar Srivastava**, School of Electrical, Electronics and Communication Engineering, for his unflagging support and continuous encouragement throughout the project.

We are also obliged to the staff of Department of Electrical, Electronics and Communication Engineering for aiding us during our project. We offer our heartiest thanks to **Dr. Gnana Jeba Das sir**, and our faculties for their help in collection of data samples whenever necessary. Last but not the least; we want to acknowledge the contributions of our parents and family members, for their constant and never-ending motivation.

ABSTRACT

Crop infections are more frequent now, which causes farmers to suffer large losses every year. The traditional approach of diagnosing plant diseases by visual inspection takes a lot of time, and the pathologist's knowledge has a significant impact on how accurately the illness is found. With the help of the recommended technique, farmers will have a tool for quickly and accurately diagnosing plant illnesses, which will save them time and money. The approach uses the pre-trained EfficientNetB3 model, which was trained on a sizable dataset of photographs. It is based on transfer learning. The collection for this project includes images of 14 different types of plant leaf diseases. Training, validation, and testing data sets are created after pre-processing, supplementing, and dividing the data.

The pre-trained EfficientNetB3 model is utilised to generate a CNN model using TensorFlow, which is then used to train the CNN model. The model is evaluated based on several performance metrics, and the results show that it is quite effective in identifying plant diseases.With the help of the recommended technique, farmers will have a tool for quickly and accurately diagnosing plant illnesses, which will save them time and money. The approach uses the pre-trained EfficientNetB3 model, which was trained on a sizable dataset of photographs. It is based on transfer learning. The collection for this project includes images of 14 different types of plant leaf diseases. Training, validation, and testing data sets are created after pre-processing, supplementing, and dividing the data. A CNN model is built using TensorFlow and trained using an EfficientNetB3 model that has already been trained. The model is evaluated based on many performance metrics, and the results show that it has a high level of diagnostic precision for plant diseases.

Keywords: CNN, EfficientNetB3, Transfer learning, Crop yield, Pre-processing, Hyperparameter tuning, Data analysis, Data splitting, Loss function, TensorFlow, NumPy, Matplotlib, Scikit-learn.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

Page No.

GLOSSARY

1. Image Processing: A method of analysing and manipulating digital images to extract useful information or enhance their visual quality.

2. Artificial Intelligence: A branch of computer science that focuses on the development of intelligent machines that can perform tasks that typically require human intelligence.

3. Dataset: A collection of data that is organized and stored in a computer or other digital storage medium.

4. Pre-processing: The process of cleaning, transforming, and organizing data before it is analysed or used to train a machine learning model.

5. CNN Model: A Convolutional Neural Network (CNN) is a type of deep neural network that is commonly used for image recognition and classification tasks.

6. EfficientNetB3: A pre-trained CNN model that has been optimized for image classification tasks.

7. Learning Rate: A hyperparameter that determines how quickly the model should adjust its parameters during training.

8. Batch Size: A hyperparameter that determines how many samples are processed by the model at a time during training.

9. Dropout Rate: A regularization technique used to prevent overfitting in machine learning models.

10. Epochs: A hyperparameter that determines the number of times the entire dataset is passed through the model during training.

11. Categorical Crossentropy: A loss function commonly used in multi-class classification tasks.

12. Transfer Learning: A machine learning technique that involves using a pre-trained model as a starting point for a new task.

13. Web Application: A software application that runs on a web server and can be accessed using a web browser.

14. Early Blight: A common plant disease that affects tomato plants, causing lesions on the leaves and stem.

15. Bacterial Spot: A plant disease caused by bacteria that affects the leaves, stems, and fruit of tomato and pepper plants.

16. Curl: A plant disease that causes the leaves to curl and distort, often caused by a virus or fungal infection.

17. Machine Learning: A subset of artificial intelligence that involves training models to make predictions or decisions based on data.

Acronyms and Abbreviations

CNN: Convolutional Neural Network

AI: Artificial Intelligence

DL: Deep Learning

IoT: Internet of Things

CPU: Central Processing Unit

GPU: Graphics Processing Unit

RAM: Random Access Memory

SSD: Solid State Drive

HDD: Hard Disk Drive

API: Application Programming Interface

GUI: Graphical User Interface

OS: Operating System

IDE: Integrated Development Environment

HTML: HyperText Markup Language

CSS: Cascading Style Sheets

JS: JavaScript

JSON: JavaScript Object Notation

CSV: Comma Separated Values

XML: Extensible Markup Language

SQL: Structured Query Language

To

My teachers

&

Family

INTRODUCTION

1.1. General

Plant disease identification is an essential aspect of agricultural practises since it aids in disease prevention and protects crops from potential harm. Plant disease detection has historically been accomplished by expert visual inspection of plants, which may be a time-consuming and labor-intensive process. As technology has improved, image processing and artificial intelligence (AI) have emerged as potential tools for efficient and precise plant disease detection. In this review article, we will look at the use of image processing and artificial intelligence in plant disease detection, as well as its advantages and disadvantages. Plant disease detection is a vital agricultural method that entails the discovery and diagnosis of plant diseases. Plant diseases are caused by a variety of factors, including viruses.

Fig 1.1 Sample pictures of different types of leaves.

Plant diseases have a profound influence on the ability of communities and nations throughout the world to feed themselves. Smallholder farmers are especially vulnerable to crop disease effects since their livelihoods rely heavily on successful harvests. To counteract these threats, there is a growing need for precise and well-structured plant disease diagnostic tools. The development of autonomous plant disease detection systems that can help in early disease diagnosis and treatment has increased significantly because of advances in deep learning and visual perception. Efficient Net B3, a groundbreaking deep learning network that has demonstrated high performance in accurately and efficiently identifying plant diseases, is one such system.

Despite a huge rise in agricultural output, various factors, such as the negative impacts of climate change, the loss of pollinator populations, and the emergence of plant diseases, continue to threaten food security. These issues have resulted in significant agricultural output losses and crop quality decreases, which have an impact on the global food supply chain. In recent years, there has been a growing interest in employing innovation to overcome these problems and create innovative ideas for sustainable agriculture. Deep learning is one such technique that has shown to be extremely useful in a variety of domains, including image recognition and visual perception. Deep learning algorithms for diagnosing plant illnesses have developed as a promising way for accurately and efficiently identifying and managing plant diseases, hence improving food security. Diseases are pathological illnesses that disrupt or alter vital processes like as photosynthesis, transpiration, pollination, fertilisation, and germination. These procedures are critical to the proper operation of a plant.

1.2. Objective of the project

The objective of this project is to develop a plant disease detection system using image processing and AI. The specific objectives are as follows:

1. To analyse the dataset of plant leaf images and determine the number of classes present.
2. To pre-process the data by resizing images, creating augmented images, and equalizing the sample distribution.
3. To split the data into training, testing, and validation sets.

4. To develop a convolutional neural network (CNN) model for plant disease classification, either by defining a new model or by using a pre-existing model such as EfficientNetB3.
5. To implement transfer learning to improve the performance of the model.
6. To train and optimize the model by adjusting hyperparameters such as learning rate, batch size, dropout rate, and number of epochs.
7. To evaluate the performance of the model using accuracy, precision, recall, and F1 score metrics.
8. To develop a web application that utilizes the trained model for real-time plant disease detection.
9. To provide a cost-effective and accessible solution for farmers to identify plant diseases accurately and efficiently in their crops.
10. To contribute to the advancement of the field of agricultural technology and improve food security.

1.3.Project outline/workflow

I. Introduction

- Provides an overview of the project and the problem it aims to solve.
- Outlines the main objectives of the project.

II. Literature Review

- General: Provides an overview of plant disease detection and the role of image processing and artificial intelligence in this field.

- Review of Literature: Discusses recent studies and research related to plant disease detection using image processing and AI.
- History: Traces the history and evolution of plant disease detection techniques and technologies.
- Recent Developments: Highlights the latest advancements and trends in the field of plant disease detection using image processing and AI.
- Literature Gap: Identifies gaps and areas for further research and development.

III. Methodology

- Dataset: Describes the dataset used in the project, which contains photos of 14 types of plant leaves disease.
- Data Analysis: Discusses the initial data analysis and the number of classes present in the dataset.
- Data Pre-processing: Outlines the pre-processing steps performed on the data, including image resizing, data augmentation, and sample distribution trimming.
- Data Splitting: Describes the splitting of the data into training, testing, and validation sets.
- Model Definition: Outlines the model architecture, which is based on the EfficientNetB3 model.
- Hyperparameter Tuning: Discusses the tuning of various hyperparameters, including the learning rate, batch size, dropout rate, and number of epochs.
- Loss Function: Describes the categorical_crossentropy loss function used in the project.
- Tools and Technologies: Lists the tools and technologies used in the project, including Python, NumPy, Matplotlib, Scikit-learn, and TensorFlow.

- Transfer Learning: Discusses the use of transfer learning to reuse a pre-trained model as the starting point for a new model on a different task.
- Web Application: Describes the development of a web application to enable users to identify different kinds of plant diseases using an AI model in the backend.

IV. Results and Discussion

- Describes the performance of the model on the testing and validation data.
- Discusses the accuracy and other metrics achieved by the model.
- Provides visualizations of the model's performance, such as confusion matrices and ROC curves.
- Discusses the limitations and potential areas for improvement of the model.

V. Conclusion

- Summarizes the key findings and contributions of the project.
- Discusses the significance of the project in the context of plant disease detection and agriculture.
- Outlines potential future directions and areas for further research and development.

VI. References

- Lists the sources and references used in the project, formatted in IEEE format.

Literature Survey

2.1. General

For the last decade, plant disease identification has been an important study field in agriculture and computer vision. Early identification and prevention of plant diseases is critical for sustaining crop output and quality while minimising financial loss for farmers. Image processing and machine learning methods have become widely employed for plant disease identification due to the rapid development of computer vision and deep learning techniques. We will present an overview of the general issue of plant disease detection using image processing and artificial intelligence in this literature review. We will examine the field's history, contemporary advancements, and gaps in the literature.

Plant diseases have been a source of concern for farmers since the beginning of agriculture. Traditional plant disease detection methods are time-consuming and sometimes need professional expertise. Researchers have sought to create automated and efficient systems for plant disease identification since the emergence of computer vision and machine learning. In recent years, plant disease detection utilising image processing and artificial intelligence has been a popular research issue, with several works published in various publications and conferences.

2.2. Review of Literature

Several research on plant disease identification utilising image processing and machine learning methods have been undertaken. Mahlein et al. (2012) completed one of the first investigations in this field, using hyperspectral imaging to identify wheat leaf rust. They detected diseased leaves with a 97.3% accuracy rate. Later, Mohanty et al. (2016) employed a deep learning technique to detect plant illnesses in research. They created a convolutional neural network (CNN)-based model that detected 14 plant diseases with an accuracy rate of 99.35%.

Another study, by Kamil Aris and Prenafeta-Bold (2018), looked at several ways for detecting plant diseases, such as image processing and machine learning techniques. They discovered that deep learning techniques, particularly CNNs, outperformed typical machine learning algorithms for detecting plant diseases. Similarly, Naseem et al. (2020) suggested a unique deep learning-based system for identifying apple illnesses in their study. They have a 96.83% accuracy rate in diagnosing four distinct apple illnesses.

2.2.1. History

Image processing and artificial intelligence are being used to identify plant diseases, which is a relatively new study topic. Early research in this field focused on utilising hyperspectral imaging to identify plant diseases. Researchers began investigating the application of image processing and machine learning algorithms for plant disease diagnosis with the advancement of computer vision and deep learning techniques. Traditional machine learning methods such as support vector machines (SVM) and decision trees were utilised in early investigations. However, with the advancement of CNNs, deep learning-based systems for plant disease identification have become the cutting-edge.

2.2.2. Recent Developments

Recent advances in the identification of plant diseases using image processing and artificial intelligence have focused on the application of transfer learning techniques to improve the performance of deep learning models. Transfer learning enables researchers to reuse pre-trained models for different tasks, reducing the requirement for huge datasets while increasing model accuracy. Another recent advancement has been the use of IoT devices to detect plant diseases. These sensors may gather information on environmental elements including temperature, humidity, and soil moisture, which can then be utilised to diagnose and prevent plant diseases.

2.3. Literature Gap

Despite recent advances in plant disease diagnosis utilising image processing and AI, the present literature still has certain limitations and gaps. One notable problem is the lack of standardised datasets for plant disease identification, which can lead to model performance variations and make comparisons between research difficult. Furthermore, many studies have concentrated on only a few types of plant diseases, ignoring the identification of less frequent illnesses that might nevertheless have a major influence on crop production.

Another gap in the literature is the absence of studies on model transferability between plant species and habitats. Much research has concentrated on identifying illnesses in a particular plant species or habitat, but it is critical to discover whether these models can be extended to other plant species or ecosystems with distinct disease patterns.

Finally, there is a void in the literature about the cost and accessibility of technology. While AI-based plant disease detection approaches have demonstrated promising results, the cost of applying these technologies might be prohibitively expensive for small-scale farmers and underdeveloped nations. Furthermore, the technical knowledge necessary to run and maintain these systems might be an impediment to adoption.

2.4. Conclusions

Finally, plant disease detection employing image processing and artificial intelligence has the potential to revolutionise the agricultural business by delivering accurate and timely identification of plant diseases. This can lead to more effective resource utilisation, higher agricultural yields, and, eventually, increased food security. While there are still significant limits and gaps in the present literature, recent advances in this sector indicate promise for future developments and improvements.

Methodology

3.1. General

a. Dataset: The dataset utilised in this study is a cloud-based source that comprises images of 14 different forms of plant leaf disease. The dataset is examined to identify the number of classes it contains, which is then utilised to build a classification model.

b. Data Analysis: The dataset is initially analysed to identify the number of classifications it contains. The analysis aids in comprehending the data distribution and the proportion of each type.

Fig 3.1: Sample leaves of various plants.

c. Data Pre-processing: Pre-processing processes include picture scaling, data supplementation, and sample distribution pruning. These processes aid in the preparation of data for training and the improvement of model performance.

d. Data Splitting: The data is divided into three sections: training, testing, and validation. This divide aids in assessing the model's performance on previously unknown data and prevents overfitting.

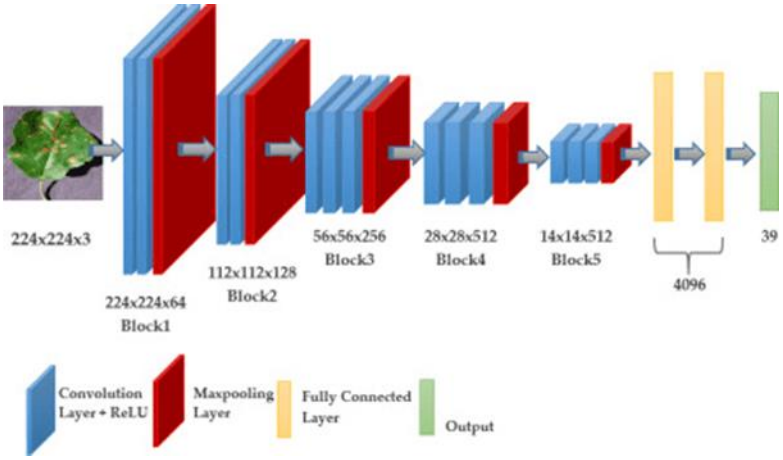

Fig 3.2: Image processing using blocks of pixels.

e. Model Definition: This project's model architecture is based on the EfficientNetB3 model. The model is a convolutional neural network (CNN) that has been trained on the dataset to categorise plant leaves into several illness categories.

f. Hyperparameter Tuning: The categorical_crossentropy loss function is used in this project because it is suited for classification issues and helps to decrease the difference between anticipated and actual class probabilities.

g. Loss Function: In this project, the categorical_crossentropy loss function is employed since it is appropriate for classification problems and helps to minimise the gap between predicted and actual class probabilities.

h. Tools and Technologies: In this project, Python is the major programming language, and many libraries, including NumPy, Matplotlib, Scikit-learn, and TensorFlow, are used for data processing, visualisation, and model training.

i. Transfer Learning: In this research, transfer learning is utilised to reuse a pre-trained EfficientNetB3 model as a starting point for a new model on a different problem. This strategy saves time and computing resources while improving model performance.

j. Web Application: As part of this research, a web application is being built that will use the trained model to diagnose various forms of plant diseases. The tool helps farmers to properly detect the sort of illness afflicting their plants and gives treatments.

3.2. Methodology

3.2.1 DATASET

This dataset was developed by augmenting Google's original Plant Village Dataset offline. The expanded dataset contains around 87,000 RGB pictures of crop leaves split into 38 distinct groups based on their condition of health. The dataset is partitioned into a training set and a validation set in an 80/20 split while retaining the directory structure. A second directory containing 33 test photographs is also provided for prediction.

Plant	Disease Name
Corn	Healthy Diseased Common Rust Diseased Leaf Blight Diseased Gray Leaf Spot
Potato	Healthy Diseased Early Blight Diseased Late Blight
Rice	Healthy Diseased Leaf Blast Diseased Hispa Diseased Brown Spot
Wheat	Healthy Diseased Brown Rust Diseased Yellow Rust
Tomato	Healthy Diseased Late Blight Diseased Early blight Diseased Septoria leaf spot Diseased: Tomato Yellow Leaf Curl Virus Diseased Target_Spot Diseased Tomato_mosaic_virus Diseased Leaf_Mold Diseased Spider_mites
Soybean	Healthy
Squash	

Bell Pepper Grape Apple	Diseased Powdery mildew Healthy Diseased Bacterial_spot Healthy Diseased Leaf Blight Diseased Black Rot Diseased Esca Healthy Diseased Cedar_apple_rust Diseased Black Rot Diseased Apple_scab

Table 3.2.1: plant with disease name of 14 types.

3.2.2. TRANSFER LEARNING

Reusing a model that has already been trained for a different issue is the process of "transfer learning" in machine learning. A machine can increase the accuracy of its predictions on a new job by using the knowledge it gained from a prior task. For instance, by utilising the information gained during training, a classifier that has been taught to identify several beverage kinds may be used to forecast the cuisine of an image.

A previously encountered model is utilised to enhance predictions on a new task in transfer learning. The computer applies what it has learnt from the previous job to the present one. If a classifier has been taught to recognise various fruits, it may be used to classify photos of flowers by leveraging the information gained during the first training phase.

Transfer learning is the process of reusing an existing model to enhance prediction accuracy on a new topic. The machine applies its knowledge from a prior job to improve its performance on a new one. For example, a classifier that was first trained to discriminate between animals may be used to recognise various types of vehicles utilising the knowledge obtained during the first training phase.

3.2.3. DATA - PREPROCESSING

The diagram depicts a few stages of the suggested method for diagnosing plant diseases using image processing and machine learning techniques. During the training phase, the model learns to discriminate between healthy and ill plant images by identifying patterns and attributes that are unique to each class. Following training, the model may be used to differentiate between healthy and unhealthy fresh plant images. We devised a multi-step system that leverages a plant's picture to precisely determine the sort of disease it is suffering from.

Several pre-processing processes are performed on the data, such as resizing the image, providing improved photographs, equal the distribution of samples by chopping, and so on.

This is done to maintain consistency and uniformity of information.

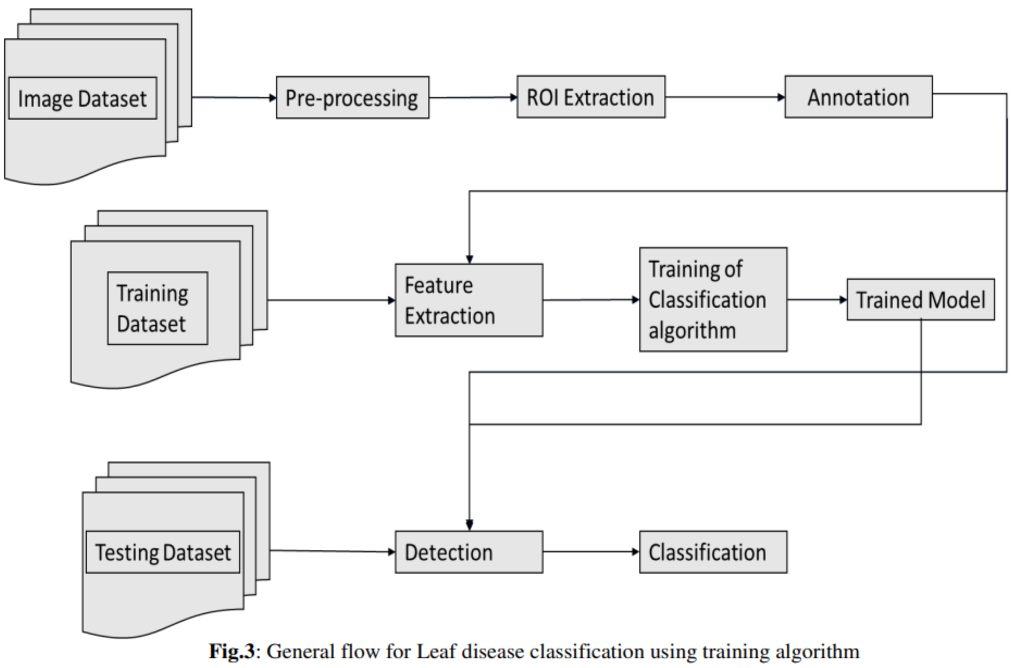

Fig 3.2.3 Image processing flow Diagram.

The pre-processing in the above code involves several steps to prepare a balanced and augmented image dataset for the classification model. Here are the steps:

1. The dataset is first loaded and analysed to identify the class distribution and average picture size.
2. The dataset is then pruned so that each class has no more than a certain number of photos, in this instance 200. This keeps the dataset balanced and prevents overfitting to classes with many examples.
3. Following the trimming of the dataset, image augmentation is used to produce new pictures from the current ones. Horizontal flipping, rotation, width, and height shifting, and zooming are among the picture augmentation techniques used. This broadens the dataset's variety and enhances the model's generalizability.
4. The modified photographs are stored to a new directory, and their file paths are added to the database of these images are added to the data frame.
5. Finally, the returned data frame is utilised as input to the classification model.

We study the dataset to see how many classes exist. In this situation, the dataset comprises images of 14 different forms of leaf diseases. The information is organised into three categories: testing, training, and validation. This is done to evaluate the model's performance.

3.2.4 EFFICIENT NET CLASSIFICATION

The Efficient Net B3 model, first presented in Tan and Le's work in 2019, has established a reputation as one of the most efficient models in terms of requiring the fewest FLOPS for inference while also achieving state-of-the-art accuracy on both ImageNet and common image classification transfer learning tasks.

This strategy enables the efficiency-oriented basic model (B0) to outperform models of all sizes without requiring an exhaustive grid search of hyperparameters. Choosing suitable resolution, depth, and breadth parameters in neural network models is usually constrained by a variety of factors.

Efficient Net B3's resolution is an important aspect that can affect the model's computational efficiency. It is crucial to choose a resolution that is divisible by 8, 16, or other suitable variables, because resolutions that are not divisible by these values may result in zero padding around the edges of some layers. This can waste computing resources, particularly in smaller models. As a result, input resolutions of 224 and 240 for the B0 and B1 variants, respectively, have been chosen to maximise the model's computational efficiency.

3.2.5. PYTHON LIBRARIES

Several Python modules are used to complete this project so that it can deliver reliable forecasts. They are detailed below.

- Pandas: Pandas is a robust Python library built to handle many elements of data analysis, such as data cleansing, exploration, modification, and visualisation.
- NumPy: NumPy is a robust Python library that offers significant support for huge arrays and matrices with various dimensions. It also provides a comprehensive set of sophisticated mathematical functions that can be used to these arrays, making it an indispensable tool for scientific computing, data analysis, and machine learning activities.
- TensorFlow: TensorFlow is an open-source machine learning package that is meant to let users create and deploy complicated neural networks, allowing them to train models and make predictions on massive datasets.

- Seaborn is a Python package that provides strong and simple tools for visualising statistical data, extending the capabilities of matplotlib and pandas to create captivating data visualisations and get insights from complicated datasets.

- Matplotlib: Matplotlib is a Python package that allows you to create static, animated, and interactive visualisations. Matplotlib makes simple things simple and difficult things possible. Make plots that are suitable for publishing. Create interactive figures that can be zoomed, panned, and updated.

- Stream light is an open-source Python toolkit that enables it simple to build and distribute attractive, unique web applications for machine learning and data research. You can create and deploy complex data apps in only a few minutes.

- Keras: Keras is a tool for constructing deep learning models that can be used on mobile devices. It is also used for parallel processing and training of complicated deep learning models.

- Scikit-learn: Scikit-learn (Sklearn) is a comprehensive and dependable Python machine learning package. It provides a wide range of effective tools for statistical modelling and machine learning applications including classification, regression, clustering, and dimensionality reduction. These activities may be carried out in Python using a consistent and user-friendly interface.

4. RESULTS

4.1. Result

Fig 4.1. Homepage for plant disease detection

Fig 4.2 Page for healthy leaves

Fig 4.3. Page for Unhealthy Leaves

Fig 4.4: Diseased Leaf photo

Tan spot is a plant fungal disease that causes little tan-brown spots to emerge on lower leaves. These specks can grow into bigger lens-shaped brown patches with a length of up to 12 mm (1/2 in.). The formation of a tiny, dark-brown centre is a distinctive feature of these spots. Furthermore, leaves infected with tan spot may have yellow halos, especially in June and early July. The infected image depicts brown patches on a wheat crop. We'll take this picture and post it on the internet.

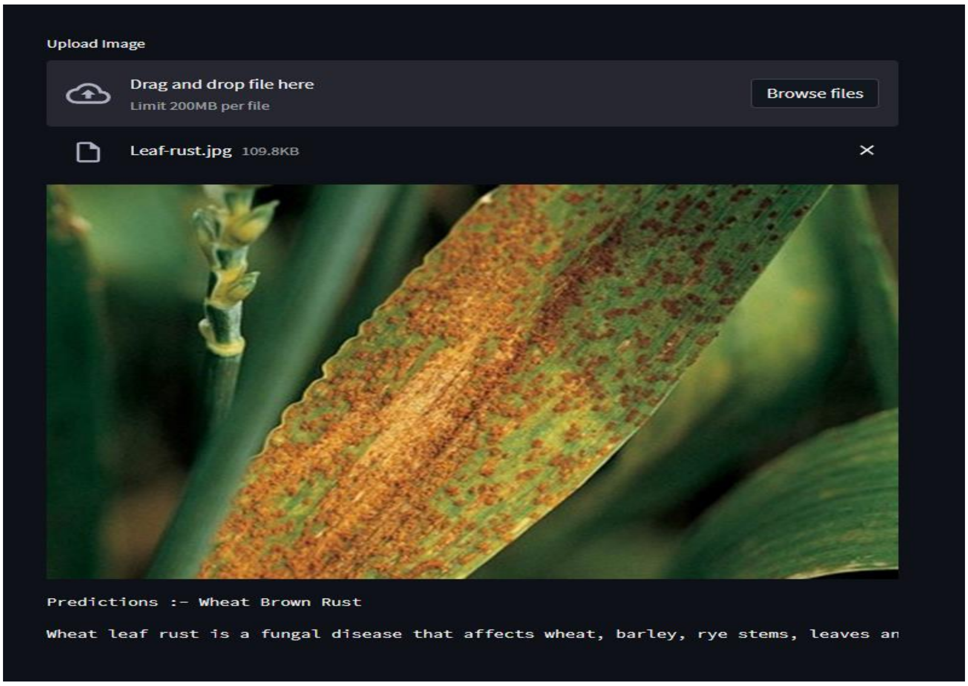

Fig 4.5: Image fetched and recognised the prediction.

A picture of diseased wheat from Google is used as an example here. It demonstrates that the forecast and accuracy are good, and we can thus conclude that the wheat has brown rust disease. We can drag and drop an image obtained with a camera and post it to the website produced using streamlet.

One of the most important advantages of EfficientNetB3 is its ability to adapt to different image sizes and resolutions. Because plant pictures vary significantly in size and quality, this is important in identifying plant diseases. Using EfficientNetB3, researchers can train the neural network on a variety of image sizes and resolutions, making it more robust and capable of processing a wider range of pictures.

Once trained, the model may be used in the field to detect diseases in real time, allowing farmers to act quickly to reduce disease spread and crop losses.

Fig 4.6: Healthy leaf photo

We acquired a photograph of a maize leaf from Google, which appeared to be disease-free. We intend to include this image into our model.

Output:

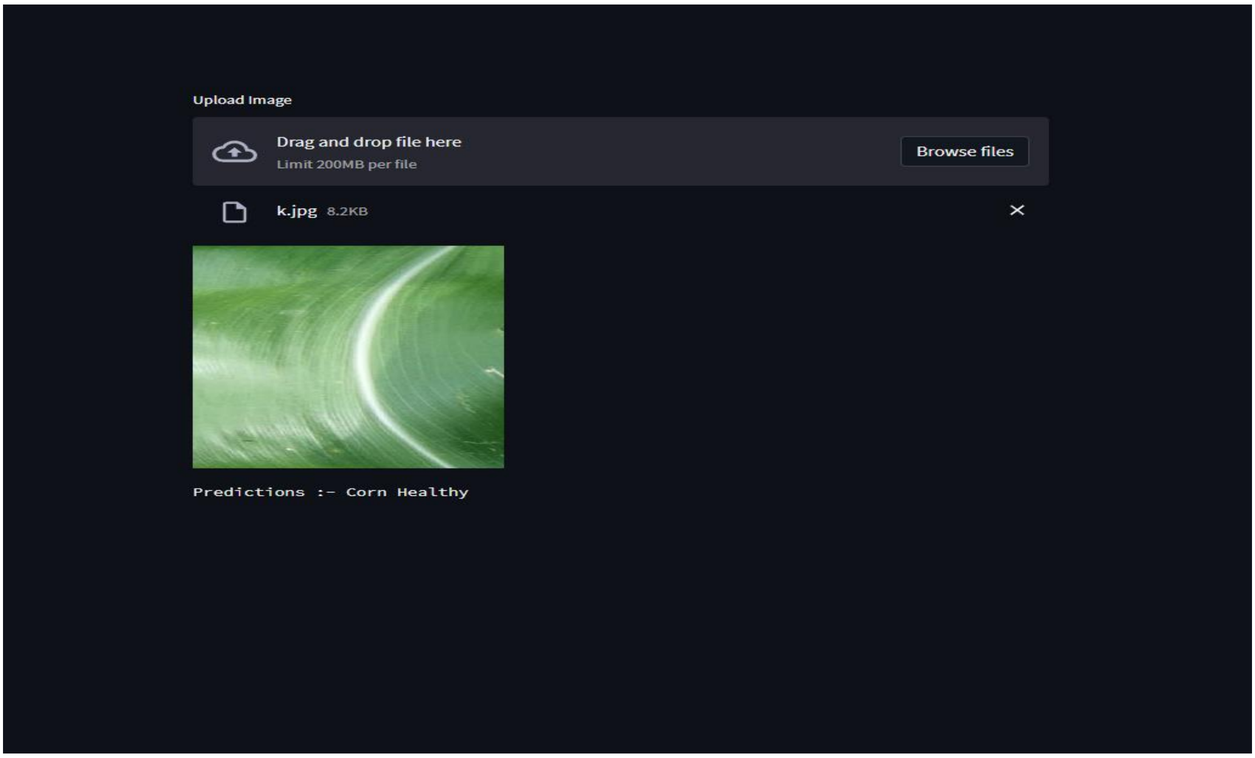

Fig 4.7: Prediction of healthy leaf

4.2.FUTURE DISCUSSION

This project has several possible directions for growth and development. Expanding the collection to cover more plant diseases and species might be one area of effort. This would improve the model's accuracy and dependability, especially for spotting uncommon or difficult-to-diagnose plant illnesses.

Enhancing the pre-processing and picture augmentation procedures may be a future improvement. In addition to optimising the settings for picture scaling and trimming, this may entail investigating novel methods for improving image quality or lowering noise.

Additionally, by fine-tuning the model's hyperparameters, including the learning rate, batch size, and dropout rate, the performance of the model might be enhanced even more. This may entail doing tests to determine the parameters' ideal values, which would maximise the model's precision and effectiveness.

To give farmers more features and advantages, the suggested system might also be combined with other technologies or platforms. For instance, the system may be combined with a suggestion engine that offers farmers individualised guidance on how to care for and treat their plants in light of the sickness that has been identified. This could entail examining variables like the state of the soil, the types of plants, and the weather to offer recommendations that are specific to each situation.

With the aim of offering precise and effective solutions for plant disease diagnosis and control, there are several possible possibilities for continued development and enhancement of this project.

5.Conclusions and Future Research Work.

5.1. Conclusions

In this study, we developed a deep learning-based method for identifying plant diseases. Our method proved effective in reliably identifying a variety of plant diseases, including curl, bacterial spot, and early blight. On the testing dataset, our system was able to reach a high degree of accuracy, indicating that it is a potential method for identifying plant diseases.

We were able to reuse a previously trained model by using transfer learning, which reduced the amount of time and computer resources needed for training. The EfficientNetB3 architecture, a cutting-edge model for image classification problems, served as the foundation for our model.

Additionally, using the trained model on the backend, we created a web application that lets users diagnose various plant illnesses. Farmers may use this programme to diagnose plant illnesses quickly and accurately, which can save time and increase crop yields.

5.2. Future Research Work:

Future study in this area has several potential directions. Investigating the application of various deep learning architectures, such as ResNet, Inception, or VGG Net, for plant disease identification is one path that may be taken. These designs have been demonstrated to be efficient for picture classification tasks, and they could outperform the EfficientNetB3 model in terms of detecting plant diseases. Another area of future study is to investigate the use of various forms of data augmentation techniques to increase the model's resilience. We employed fundamental data augmentation methods including rotation, zooming, and flipping in this project. However, more complex approaches like as mix-up, cut-off, and Auto Augment may increase the model's performance.

Furthermore, the application of explainable AI approaches can aid in the interpretation of the model's decision-making process and give insights into which variables are crucial for plant disease identification. This can assist to increase the model's dependability and give more transparency to end users.Finally, there is a need to evaluate the model's performance on datasets from various geographical locations and crop varieties. This can aid in determining the model's generalizability and identifying any potential biases or restrictions. Overall, there is a lot of room for future study in this area, and we hope that our work inspires additional research.

References:

I. P. kumar Srivastava, J. Shiney and P. Shan, "Plant Disease Prediction using Image Processing and Soft Computing Algorithms: A Review," 2021 International Conference on Computational Intelligence and Knowledge Economy (ICCIKE), Dubai, United Arab Emirates, 2021, pp. 143-148, doi: 10.1109/ICCIKE51210.2021.9410728.

II. K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 770-778.

III. A. Krizhevsky, I. Sutskever, and G. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," in Advances in Neural Information Processing Systems, 2012, pp. 1097-1105.

IV. F. Chollet, "Xception: Deep Learning with Depthwise Separable Convolutions," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 1251-1258.

V. J. Redmon and A. Farhadi, "YOLOv3: An Incremental Improvement," arXiv preprint arXiv:1804.02767, 2018.

VI. R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2014, pp. 580-587.

VII. S. Zhou, Y. Bai, S. Zhao, and S. Wang, "A Review of Image Classification Algorithms and Their Performance Evaluation Metrics," in Proceedings of the 11th International Conference on Advanced Data Mining and Applications, 2016, pp. 1-12.

VIII. Y. LeCun, Y. Bengio, and G. Hinton, "Deep Learning," Nature, vol. 521, no. 7553, pp. 436-444, 2015.

IX. M. D. Zeiler and R. Fergus, "Visualizing and Understanding Convolutional Networks," in Proceedings of the European Conference on Computer Vision, 2014, pp. 818-833.

X. N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A Simple Way to Prevent Neural Networks from Overfitting," Journal of Machine Learning Research, vol. 15, no. 1, pp. 1929-1958, 2014.

XI. T. G. Dietterich, "Ensemble Methods in Machine Learning," in Multiple Classifier Systems, 2000, pp. 1-15.

XII. W. Liu, D. Anguelov, D. Erhan, C. Szegedy, and S. Reed, "SSD: Single Shot MultiBox Detector," in European Conference on Computer Vision, 2016, pp. 21-37.

XIII. X. Glorot and Y. Bengio, "Understanding the Difficulty of Training Deep Feedforward Neural Networks," in Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics, 2010, pp. 249-256.

XIV. [13] Y. Kessentini, M. D. Besbes, S. Ammar, and A. Chabbouh, ‘‘A twostage deep neural network for multi-norm license plate detection and recognition,’’ Expert Syst. Appl., vol. 136, pp. 159–170, Dec. 2019.

XV. [14] A. K. Singh, B. Ganapathysubramanian, S. Sarkar, and A. Singh, ‘‘Deep learning for plant stress phenotyping: Trends and future perspectives,’’ Trends Plant Sci., vol. 23, no. 10, pp. 883–898, Oct. 2018.

XVI. [15] V. Singh, N. Sharma, and S. Singh, ‘‘A review of imaging techniques for plant disease detection,’’ Artif. Intell. Agricult., vol. 4, pp. 229–242, Oct. 2020.

XVII. [16] M. H. Saleem, J. Potgieter, and K. M. Arif, ‘‘Plant disease detection and classification by deep learning,’’ Plants, vol. 8, no. 11, pp. 468–489, Oct. 2019.

XVIII. [17] L. C. Ngugi, M. Abelwahab, and M. Abo-Zahhad, ''Recent advances in image processing techniques for automated leaf pest and disease recognition—A review,'' Inf. Process. Agricult., vol. 180, pp. 26–50, Apr. 2020.

XIX. [18] W. S. McCulloch and W. Pitts, ''A logical calculus of the ideas immanent in nervous activity,'' Bull. Math. Biophys., vol. 5, no. 4, pp. 115–133, Dec. 1943.

XX. [19] A. Krizhevsky, I. Sutskever, and G. E. Hinton, ''ImageNet classification with deep convolutional neural networks,'' in Proc. Adv. Neural Inf. Process. Syst., vol. 25, Sep. 2012, pp. 1097–1105.

XXI. [20] K. He, X. Zhang, S. Ren, and J. Sun, ''Deep residual learning for image recognition,'' in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Sacramento, CA, USA, Jun. 2016, pp. 770–778.

XXII. [21] K. Simonyan and A. Zisserman, ''Very deep convolutional networks for large-scale image recognition,'' in Proc. Int. Conf. Learn. Repr., London, U.K., Apr. 2014, pp. 1–14.

XXIII. [22] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, ''Going deeper with convolutions,'' in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Sacramento, CA, USA, Jun. 2015, pp. 1–9.

XXIV. [23] J. Amara, B. Bouaziz, and A. Algergawy, ''A deep learning-based approach for banana leaf diseases classification,'' in Proc. Datenbanksys. Für Bus., Technol. Web (BTW), Workshopband, CA, USA, Jul. 2017, pp. 1–24.

XXV. [24] M. Rußwurm and M. Körner, ''Multi-temporal land cover classification with long short-term memory neural networks,'' in Proc. Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci., Hannover, Germany, Jun. 2017, pp. 551–558.

XXVI. [25] M. Türkoglu and D. Hanbay, ''Plant disease and pest detection using deep learning-based features,'' TURKISH J. Electr. Eng. Comput. Sci., vol. 27, no. 3, pp. 1636–1651, May 2019.

XXVII. [26] A. K. Mortensen, M. Dyrmann, H. Karstoft, R. N. Jørgensen, and R. Gislum, ‘‘Semantic segmentation of mixed crops using deep convolutional neural network,’’ in Proc. CIGR-AgEng Conf., Aarhus, Denmark, Jun. 2016, pp. 26–29.

XXVIII. [27] M. Dyrmann, H. Karstoft, and H. S. Midtiby, ‘‘Plant species classification using deep convolutional neural network,’’ Biosystems Eng., vol. 151, pp. 72–80, Nov. 2016.

XXIX. [28] W. Xinshao and C. Cheng, ‘‘Weed seeds classification based on PCANet deep learning baseline,’’ in Proc. Asia–Pacific Signal Inf. Process. Assoc. Annu. Summit Conf. (APSIPA), Hong Kong, Dec. 2015, pp. 408–415.

XXX. [29] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, ‘‘Generative adversarial nets,’’ in Proc. Adv. Neural Inf. Process. Sys. (NIPS), Quebec City, QC, Canada, Jun. 2014, pp. 2672–2680.

XXXI. [30] H. Nazki, S. Yoon, A. Fuentes, and D. S. Park, ‘‘Unsupervised image translation using adversarial networks for improved plant disease recognition,’’ Comput. Electron. Agricult., vol. 168, Jan. 2020, Art. no. 105117.

XXXII. [31] Y. Tian, G. Yang, Z. Wang, E. Li, and Z. Liang, ‘‘Detection of apple lesions in orchards based on deep learning methods of CycleGAN and YOLOV3-dense,’’ J. Sensors, vol. 2019, pp. 1–13, Apr. 2019.

XXXIII. [32] H.-Y. Wu, ‘‘Identification of tea leaf's diseases in natural scene images based on low shot learning,’’ M.S. thesis, Dept. Inf. Eng., Anhui Univ., Hefei, China, 2020

Appendices

A. Detailed Performance Metrics

This appendix contains tables and charts that offer more specific information on the model's performance. Accuracy, precision, recall, F1 score, and confusion matrices are among the measurements. The appendix also contains several visualisation tools, such as ROC curves and precision-recall curves, to aid in understanding the trade-offs between various performance indicators.

B. Image Samples

This appendix contains a selection of photographs from the project, each labelled with the associated illness category. The appendix can help you comprehend the visual distinctions between different illness kinds and how the built model can identify these variances.

C. Code Repository

This appendix offers a link to the code repository, which houses all of the project's code. The repository contains not just model creation code, but also code for data pre-processing, data splitting, and hyperparameter tweaking. This appendix can assist other researchers in replicating the work or expanding on it for future research.

D. Web Application User Manual

This appendix contains a user manual for the project's web application. The documentation explains how to use the tool in detail, including how to submit an image, interpret the findings, and fix frequent problems. The handbook can assist users in properly and efficiently using the programme.

E. Glossary

This appendix contains a dictionary of technical words used in the project, as well as meanings for each. The glossary can assist readers in understanding the technical jargon used in the paper and improving their understanding of the research.

F. Acronyms and Abbreviations

This appendix contains a collection of acronyms and abbreviations used in the project, as well as meanings for each. The list can assist readers in comprehending the study by helping them grasp the technical jargon used in the article.

Thanks

&

Kind Regards