International Research Journal of Education and Technology
Peer Reviewed Journal
ISSN 2581-7795

PLANT LEAF DISEASE PREDICTION USING EDGE AI

Mrs. S. Sabeena, Assistant Professor,

Department of Software Systems, Sri Krishna Arts And Science College, Coimbatore,

Tamil Nadu, India

Aswin V, Student, Department of Software Systems, Sri Krishna Arts and Science College,

Coimbatore, TamilNadu, India

Abstract

This paper provides an Edge AI powered framework for predicting plant leaf diseases in real-time applications of precision farming is discussed. The suggested model involves a compact CNN architecture hosted on the edge device to conduct inference operations. In contrast to existing cloud computing solutions that rely on transferring data to the server, the proposed model conducts image processing operations locally, which minimizes response time and reduces the need for internet connectivity. The CNN model is trained on a labeled database of plant leaves images and is further fine-tuned through methods like quantization and pruning. Experimental analysis validates the efficacy of the model, with high accuracy rates achieved under minimal latency conditions.

Keywords

Edge AI, Leaf Disease Detection, Smart Agriculture, CNN, Deep Learning, Image Processing, IoT

I. Introduction

The sector of agriculture is fundamental when it comes to ensuring food security and economic stability in developing areas.

There are a number of factors that significantly impact the productivity of the process, including diseases occurring in plants, particularly in their leaves. This is because leaf state often corresponds to general plant condition and is important when it comes to disease detection.

Deep learning techniques have been developed to help detect various diseases on images. In particular, CNNs have proved to be useful in the detection of such symptoms as discoloration, spots, and changes in texture. However, most current approaches focus on cloud-based computations, which may lead to certain delays and problems related to connectivity in less developed areas.

However, in an effort to overcome these problems, there comes an innovative solution of using edge computing. By implementing this technology, decisions can be made faster as well as reduce dependency on other infrastructure. In this regard, combining edge AI with the detection of plant diseases can be considered as one of the effective ways of doing so.

In this research paper, I intend to develop an efficient edge AI system that can detect different leaf diseases through classification.

II. Literature Review

Several computational methods for detecting diseases in plants' leaves have been proposed, with image processing techniques being applied from basic to advanced ones based on Deep Learning. Traditional image processing techniques have used hand-crafted feature extraction methods including color histogram extraction, texture analysis, and edge detection. The extracted features would be used with machine learning algorithms to classify the image data into different classes, including SVM and k-Nearest Neighbor (k-NN) methods. Although computationally fast, the aforementioned methods do not provide sufficient performance under different conditions like light variation and background cluttering.

However, since the development of deep learning technologies, CNN techniques have been extensively implemented for developing effective classifiers for plant disease detection systems. Mohanty et al. [1] showed that CNN-based classifiers can yield excellent results in classifying images from the PlantVillage database using models like AlexNet and GoogleNet. Nonetheless, the problem with such models is that they demand much computational power and cannot be deployed on low-power hardware.

To overcome computational limitations, several lightweight CNN models were suggested. In the work by Howard et al. [2], MobileNet was designed through employing depthwise separable convolution filters that minimize the number of computations while preserving efficiency. Another architecture is EfficientNet that is designed to scale efficiently in network dimensions to increase performance while reducing the number of parameters [3]. These models would be more effective for real-time inference.

Recent literature has concentrated on the deployment of plant disease prediction models using edge computing. The advantage of the technique is that it can provide low-latency data processing without reliance on the cloud. The authors of the paper published by Shiet al. [4] pointed out the advantages of edge computing for data-intensive and low-latency application processing. Several papers have employed CNN models using TensorFlow Lite and ONNX Runtime frameworks.

However, several issues exist in trying to create a system that has optimal performance based on accuracy, efficiency, and energy consumption. Some current models cannot manage to work under real-time conditions such as uneven lighting, occlusion, and environmental noises. It is therefore crucial to develop a system that has accurate performance but is at the same time efficient.

Author / Year	**Method Used**	**Accuracy**
Mohanty et al. (2016)	AlexNet, GoogleNet	~99%
Sladojevic et al. (2016)	CNN-based model	~96%
Ferentinos (2018)	Deep CNN	~99%

Howard et al. (2017)	MobileNet	~90–95%
Tan & Le (2019)	EfficientNet	~96–98%
Recent Edge AI Studies	MobileNet + TFLite	~92–96%

Table 1. Comparison of Existing Methods

III. Problem Statement

Despite advances made thus far, plant diseases remain a major source of threat to farm productivity by reducing crop yields and generating losses. Early and accurate diagnosis of such plant diseases is necessary to minimize their spread as well as provide effective means of treatment. Unfortunately, despite the existence of several automated approaches, disease diagnosis in the real world remains manual, and thus prone to inaccuracies as well as inefficient.

Currently existing artificial intelligence-based approaches provide greater levels of accuracy in disease diagnosis, yet they rely on cloud computing infrastructure. Such systems require constant connectivity as well as increased processing time, making them ineffective in areas where network coverage may be limited.

Besides, managing large amounts of image data on a cloud-based system is problematic as it requires high bandwidths as well as higher processing times. Variations in external factors such as lighting conditions complicate disease diagnosis. This project seeks to provide a viable solution to these problems by introducing an accurate and efficient disease diagnosis system.

IV. Proposed Methodology

The proposed approach involves designing an Edge AI-based leaf disease prediction system which aims to predict plant diseases by analyzing leaf images in real-time, without the need for cloud computing facilities.

As opposed to cloud-computing-based systems which are commonly used in agricultural applications, the proposed system will be based on on-device inference, thereby ensuring reduced latency and bandwidth usage, apart from minimizing the need for internet access.

The proposed system can detect diseases in their initial stages and provide feedback to the farmers immediately. The primary aim of designing this system is to ensure rapid disease detection and diagnosis with minimum expenditure.

4.1 Overall system design

The proposed system follows a structured **four-stage pipeline**, ensuring efficient processing and accurate predictions:

- Leaf Image Acquisition
- Image Preprocessing and Feature Extraction
- Disease Prediction using Edge AI Model
- Result Interpretation and Recommendation

Each stage transforms raw image data into meaningful agricultural insights.

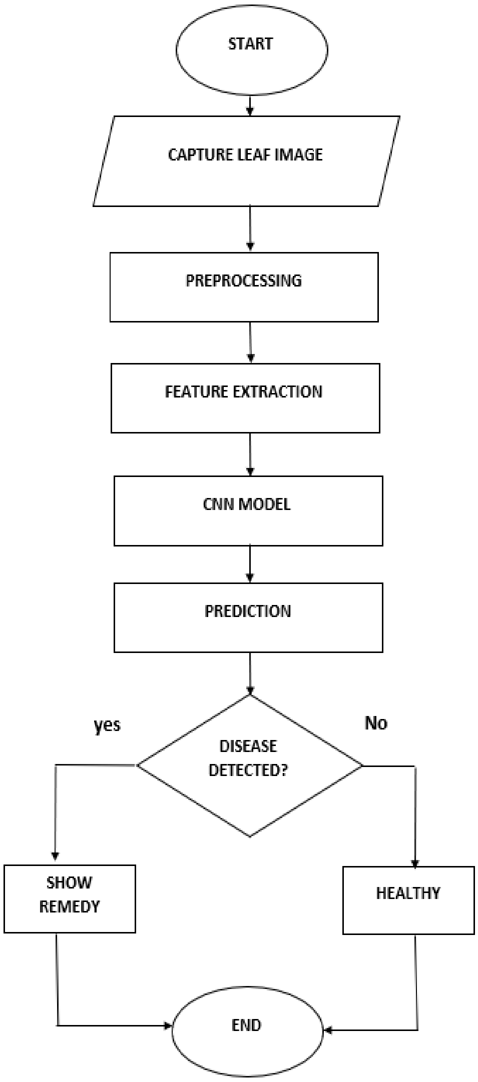

Fig 1. Workflow of the system

4.2 Leaf image acquisition

The data collection in this phase is done with the help of:

- Camera phones
- Camera modules of Raspberry Pi
- Field sensors of IoT

Data collection takes place with changing background, lighting conditions, and camera angles. This helps in generalizing the system in realistic conditions.

The system works for multiple crops and can be used with other plants by adding data points to the dataset.

4.3 Image preprocessing and feature extraction

Input images taken from the real world might suffer from noise, poor illumination, and unnecessary background information. Consequently, preprocessing will be done to enhance input quality.

Preprocessing involves:

- Resizing: Resizing the image into a fixed size (for example 224 × 224)
- Normalization: Adjusting pixel values to fall within 0 and 1
- Noise Filtering: Applying filters to eliminate distortions that cause noises
- Background Elimination: Using segmentation algorithms to segment only the plant leaf part

Feature extraction

In this project, the process of designing manually the features for classification purpose is automated using CNN that extract:

- Color features (discoloration, disease spots)
- Texture features (lesion, roughness)
- Shape features

These features are represented in form of feature map.

4.4 Edge AI based Disease Prediction

The preprocessed leaf image is then fed to a deep learning-based model on an edge computing system.

Class Label	Description
Healthy	No visible disease symptoms
Leaf Spot	Small brown or black spots on leaves
Blight	Rapid drying and decay of leaves
Powdery Mildew	White powder-like coating on surface

Table 2. Disease Classes

In the presented approach, we have used MobileNetV2 since it is highly efficient and applicable for devices with limited computing resources. MobileNetV2 uses depth-wise separable convolution layers that greatly reduce the number of parameters and computational cost without sacrificing accuracy. The input dimensions of the model are 224×224 pixels, and the output is provided as probability distributions over several disease classes by applying a Softmax layer.

4.5 Dataset Description

For training our proposed model, we have used the popular dataset Plant Village. This dataset consists of thousands of plant leaf images belonging to multiple categories of healthy plants and those affected with various diseases.

There are different disease classes present in the dataset such as leaf spot, blight, and powdery mildew along with a class for healthy leaves.

Class Label	Description	Number of Images
Healthy	No visible disease	1500
Leaf Spot	Small dark or brown spots	1200
Blight	Rapid drying and browning of leaves	1100
Powdery Mildew	White powder-like substance	1000
Rust	Orange or reddish patches	900

Table 3. Sample Dataset Table

4.6 Model Working

Firstly, the image is resized and then subjected to the MobileNetV2 model. The model automatically picks up on key features like color contrast, texture, and lesions of disease. These are further fed into several other layers until finally the output layer gives us the classification of that image into one of the known categories.

4.7 Model Optimization for Edge Deployment

To ensure efficient execution on edge devices, the trained model is optimized using:

Quantization

- Converts model weights from float32 → int8
- Reduces memory usage and increases speed

Pruning

- Removes less important connections
- Reduces model size without major accuracy loss

TensorFlow Lite Conversion

- Converts model into edge-compatible format

These optimizations ensure that the system runs smoothly on **low-power devices**.

4.8 Algorithm Description (CNN)

The system to be used will utilize a CNN model based on the MobileNetV2 model for the identification of plant leaf diseases. Convolutional neural networks represent deep learning techniques and are very efficient when it comes to analyzing images since they independently learn the important attributes of the images such as colors and shapes.

A CNN model involves processing an image through a series of layers where each layer has specific roles like feature extraction, pooling, and classification. This will be followed by the application of the Softmax function.

Pseudo Code (CNN)

1. Start
2. Capture input image I
3. Resize image I to 224 × 224 pixels
4. Normalize pixel values of I to range [0,1]
5. Load trained MobileNetV2 model M
6. Pass preprocessed image I into model M
7. Extract features using convolutional layers
8. Apply pooling layers to reduce feature dimensions
9. Flatten feature maps into a feature vector
10. Pass feature vector through fully connected layers
11. Apply Softmax function to obtain class probabilities
12. Identify class C with highest probability
13. Display predicted class C and confidence score
14. End

V. System Architecture

The system architecture will have the following three layers:

- Sensing Layer
- Edge Layer
- Application Layer

In the sensing layer, cameras/sensors will take pictures of the leaves. These devices will be installed on farms for continuous monitoring of their crops.

The edge layer will perform computations using an embedded system like Raspberry Pi, NVIDIA Jetson Nano, or any smartphone. Here, the model will be used to detect diseases from the input images.

In the application layer, farmers will be able to interact with the system through a graphical user interface (GUI). The GUI will show what diseases have been detected along with suggested treatment. Very little communication will occur between layers

International Research Journal of Education and Technology
Peer Reviewed Journal
ISSN 2581-7795

for less bandwidth consumption. The optional use of cloud computing can be considered for storing the data and updating the model periodically.

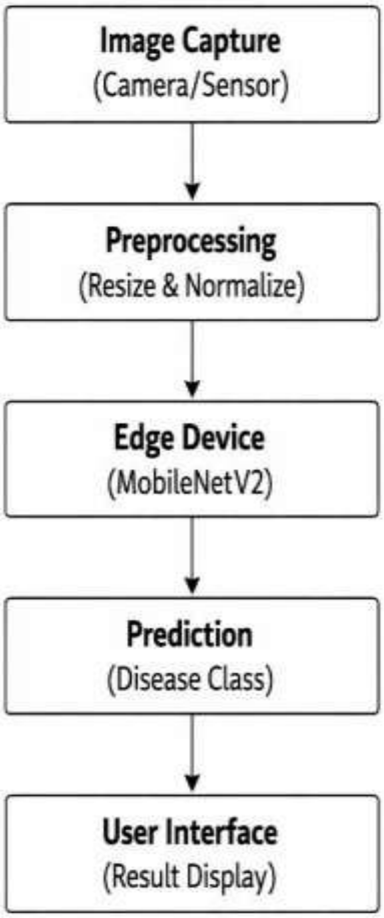

Fig 2. System Architecture

VI. Results and Analysis

Performance evaluation is done using common classification metrics such as accuracy, precision, recall, and F1 score. These measures offer a thorough analysis of the model's proficiency in predicting healthy and diseased leaves.

The algorithm is built based on the MobileNetV2 neural network architecture. It uses the PlantVillage dataset for training and testing purposes. According to the experimental outcomes, the model exhibits a very high level of classification performance and can be applied to practical agricultural settings.

The accuracy measure is 95.8%, implying that the model efficiently recognizes whether an input image corresponds to a healthy or diseased leaf sample. Moreover, the precision parameter is 94.6%. This result suggests that the algorithm minimizes the probability of predicting that a healthy leaf is diseased. In addition, the recall metric is 93.9%. This figure implies that most disease cases are detected. Finally, the F1-score of 94.2% highlights a balance between precision and recall.

Metric	Value
Accuracy	95.8
Precision	94.6
Recall	93.9
F1-score	94.2

Table 4.Performance metrics of the system

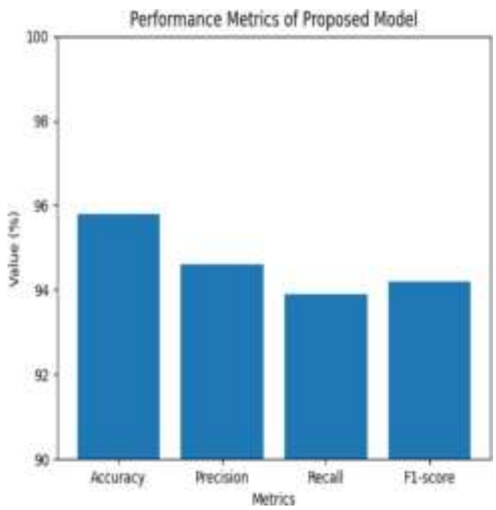

Fig 3. Performance Metrics Graph

International Research Journal of Education and Technology
Peer Reviewed Journal
ISSN 2581-7795

6.1 Latency Analysis

This model conducts its inferencing at the edge itself, without the need to go through the cloud process. It takes about 0.65 seconds per image, which is much faster than cloud computing processes that might take between 2-5 seconds each time. This will enable farmers to make more informed decisions quickly.

6.2 Resource Efficiency

The employment of MobileNetV2 together with optimization of the model through quantization leads to decreased computation needs. The system requires fewer resources for storing data and processing data, thereby making it viable for implementation in low-powered systems like the Raspberry Pi.

6.3 Input and Output Results

In order to prove the effectiveness of our system, some samples of input and output images have been given below. Our system receives an image of a leaf as input, which is classified by the pre-trained MobileNetV2 model.

Before feeding our input image into the model, we preprocess it and obtain the output in the form of the disease label and the corresponding probability score.

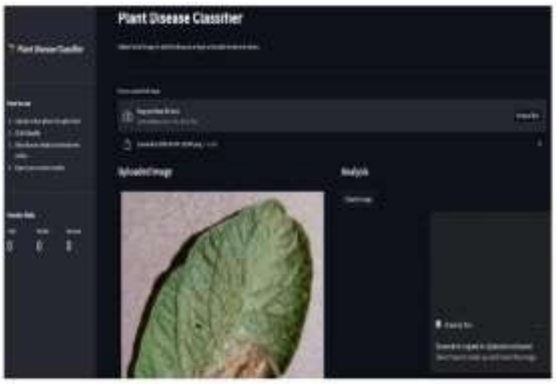

Fig 4. Input Leaf Prediction

Moreover, Figure 5 represents yet another instance in which the model has been able to successfully detect a sample of a healthy leaf. The above findings confirm that the designed algorithm is indeed able to classify both healthy and unhealthy leaves.

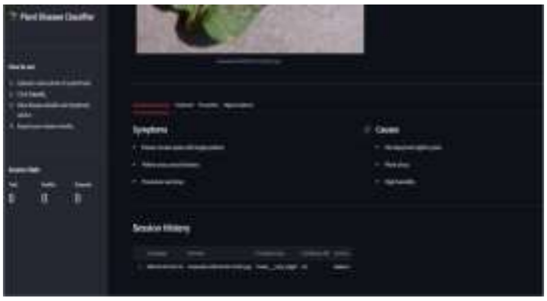

Fig 5. Output Prediction

Overall, the input-output examples validate the practical applicability of the system in real-time agricultural scenarios.

VII. Discussion

This system effectively shows how Edge AI can be used in the smart farming domain. It eliminates the need for internet connection and ensures better privacy because the processing of data is done locally. Lightweight models allow deploying this system even on inexpensive equipment which makes it affordable to small farmers.

The possibility of connecting this system with IoT can significantly improve its efficiency as well as automate monitoring. This system can potentially be developed to identify other conditions of plants including lack of nutrients. Nevertheless, model updating at regular intervals is necessary in order to ensure its proper functioning.

VIII. Conclusion

This paper provided an effective and efficient method for predicting plant leaf diseases through the implementation of edge artificial intelligence. This system used lightweight machine learning methods deployed on edge devices to perform real-time leaf disease detection immediately from the point of data capture. This process removes any dependency on cloud resources thereby reducing latency, bandwidth consumption, and operational costs.

The implementation of convolutional neural networks (CNNs) together with edge technology has led to efficient classification of leaf diseases by recognizing features such as colors and textures. Model optimization through the processes of quantization and pruning improves the efficiency of CNNs and makes the proposed system highly compatible with resource-constrained systems.

The experimental findings have shown that the proposed system performs well with high levels of accuracy and efficient performance on edge platforms. The proposed system is especially advantageous for agricultural activities performed in remote regions which do not have stable access to the Internet.

IX. Future Work

Despite the effectiveness of the proposed system for detecting leaf diseases in real time, several possible improvements could be considered in future research.

First, the use of real-time Internet-of-Things-based monitoring systems can help monitor the growth of plants continuously and send alerts in case of any problems.

Second, increasing the number of images used in training can help increase the ability of the system to generalize from diverse scenarios.

In addition, more advanced deep learning methods, such as Vision Transformers and hybrid methods, could be used to extract better features.

Furthermore, using XAI methods will allow one to explain how the prediction process takes place, making the proposed model more understandable for farmers.

Third, building a user interface on mobile applications can help users interact with the model more easily and access it regardless of their location. Finally, integrating information about the weather and using predictive analytics can help predict the spread of disease outbreaks

References

[1] S. P. Mohanty, D. P. Hughes, and M. Salathé, “Using deep learning for image-based plant disease detection,” Frontiers in Plant Science, 2016.

[2] A. G. Howard et al., “MobileNets: Efficient convolutional neural networks for mobile vision applications,” 2017.

[3] M. Tan and Q. Le, “EfficientNet: Rethinking model scaling for convolutional neural networks,” 2019.

[4] W. Shiet al., “Edge computing: Vision and challenges,” IEEE Internet of Things Journal, 2016.

[5] S. Sladojevic et al., “Deep neural networks based recognition of plant diseases,” Computational Intelligence and Neuroscience, 2016.

[6] K. P. Ferentinos, “Deep learning models for plant disease detection and diagnosis,” Computers and Electronics in Agriculture, 2018.