IJARSCT

ISSN (Online) 2581-9429

International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)

Impact Factor: 6.252

Volume 3, Issue 1, January 2023

Plant Leaf Disease Detection using IoT, DL and ML

Y. Harshavardhan Reddy[1], J. Sreekaree[2], S. Md. Ashwak[3], K. Lakshmi Pavan Kumar Sarma[4], Adnan Ali[5], A. Zaheer Sha[6], R. Varaprasad[7]

G. Pullaiah College of Engineering and Technology, Kurnool, Andhra Pradesh, India[1,2,3,4,5,6,7]

Abstract: *Agriculture is crucial in the development of the country. It takes the major role in the economy of the country. The major problem arises in the agriculture is the plant diseases. Due to heavy rains and use of the pesticides and global warming many types of disease are born and infected the crops. These plant diseases lead to the death of the crop at early stage. Various detection methods are introduced for plant diseases has some draw backs in them. Whereas plant disease detection methods are made by using IoT, ML and DL algorithms. In this paper, to overcome the problems in the previous methods. Our model is made by combining the IoT, ML and DL, by placing the algorithms in three different stages to get the higher accuracy in the plant disease detection. And end the disease at the early stage. By using three classes of tomato plants, system for plant disease detection using IoT, ML and DL was developed to predict the disease at early stage in stage-by-stage testing.*

Keywords: Internet of things, CNN, Random Forest, Agriculture

I. INTRODUCTION

Agriculture is a crucial component of the Indian economy. Seventy percent or more of rural households depend on agriculture as their primary source of income. Agriculture is a vital component of the Indian economy, accounting for approximately 17% of total GDP and serving as the primary source of income for approximately 58% of the working population. In recent decades, India's agriculture industry has experienced extraordinary expansion. Since the country's independence in 1950–1951, foodgrains production has reached a record high of 250 million metric tonnes (MT) in 2011–2012. Agriculture's contribution to the gross domestic product increased from 17.8 percent in 2019–2020 to 19.9 percent in 2020–21[1]. In the 2003–2004 fiscal year, the agriculture sector's contribution to the nation's gross domestic product was 20 percent. After witnessing a drop due to the pandemic, it is anticipated that consumer expenditure in India will begin to rise again in 2021, with growth of up to 6.6%. India's food sector is on the verge of significant expansion, which will result in a larger annual contribution to the global food trade because of India's vast potential for value addition, notably in the food processing industry. It is the fifth largest industry in India in terms of production, consumption, exports, and anticipated growth [2].

The Indian food processing industry represents 32% of the country's total food market. Plant diseases reduce the number and quality of food, fibre, and biofuel crops at a time when agriculture is trying to keep up with a rapidly expanding global population. Losses can be catastrophic or chronic, but they still account for an average of 42% of the production of the six crops regarded as the most vital to human sustenance. Using internet-of-things sensors to monitor plants and livestock, which is now being researched, is a promising method for achieving PA [3]. Internet of Things devices contain low-power embedded circuits that can communicate with other Internet of Thing's devices across a network. Individual devices communicate and collaborate to achieve a common purpose in Internet of Things-connected networks. An agricultural system based on the Internet of Things has the potential to collect environmental data, such as the soil's moisture level, using sensors. The measured data can then be used to manage an automated irrigation system to sufficiently water plants while minimising over- and under-watering. By utilising technology connected to the Internet of Things, farmers may monitor their crops remotely and in real time.

Monitoring the livestock was equally as vital as monitoring the crops in the field, and there is a correlation between monitoring the livestock and a reduced chance of disease transmission between animals and humans [4]. Utilizing Internet of Things devices can greatly improve animal welfare while simultaneously reducing labour expenses. IoT devices can provide information regarding the location and health of animals.

In this investigation, IoT, ML, and DL are combined to diagnose plant diseases [5]. For greater precision, the detection of plant diseases occurs in two phases. In Section 2, you'll find a list of the hardware and software requirements for the project. In Section 3, the specifics of the algorithms used in the research are discussed. In Section 4, the experimental apparatus is detailed. In Section 5, the findings and interpretations are provided.

II. LITERATURE SURVEY

Detection of plant disease methods are implemented more in the past using various technology mainly by using image processing and deep learning techniques. This process continues to attract a lot of researchers in this field to design different models. Plant disease detection using CNN gain more attention for the researchers. Geetha et al., (2020) [6] proposed the plant leaf disease detection using machine learning. Image pre-processing is added to remove noise, segmentation to detect the affected area on the leaf and k-nearest neighbours (KNN) algorithm is used for classification of the disease. Gayathri et al., (2021) [7] used the internet of things (IoT) and Machine learning algorithms such as SVM and CNN monitor and detect the crop disease. This model performs a comparative analysis of SVM, CNN, naive bayes, and KNN. Jun Liu and Xue wei Wang (2021) [8] proposed model using digital signal processing and image processing using CNN. CNN algorithms are used to do feature extraction. The evaluation of the model using evaluation metrics such as precision, recall, mean average precision, and mean F1 score. Vijay Singh and A.K. Misra (2016) [9] proposed model for plant leaf disease detection using image segmentation and soft computing techniques. They proposed the model with algorithm named K-Means clustering for image segmentation. They used banana, beans, jackfruit, lemon, mango, potato, tomato, and sapota plant leaves as samples for plant disease detection. Sunil et al., (2022) [10] proposed plant leaf disease detection model using computer vision and machine learning algorithms. This paper is based on detection of plant disease on tomato plants. Histogram equalization and K-means clustering are used to increase the quality of image. K. Padmavathi and K. Thangadurai (2016) [11] did a comparative study on the implementation of RGB and Grayscale images for plant leaves disease detection. They analysed the Grayscale and RGB images using techniques such as pre-processing, segmentation, and classification.

III. PROPOSED MODEL

To diagnose the disease and offer a comprehensive assessment of the sickness to which the crop had been exposed, a prototype was created. Our prototype's overall power consumption was reduced by regulating the camera module's power consumption [12]. We utilised an esp32 board as our microcontroller. DHT11 sensor, soil moisture sensor, and ESP32 camera are employed as sensors. We use the CNN algorithm to determine the type of plant disease present. In order to efficiently train the model on a personal computer using the CNN method, a graphics processing unit is required.

3.1 Hardware Requirements

A. NODEMCU

NodeMCU is an open-source Internet of Things platform that is available at a minimal cost. It initially consisted of both software and hardware, the former of which was based on the ESP-12 module and the latter of which ran on Espressif Systems' ESP8266 Wi-Fi System-on-Chip (SoC). Support for the ESP32 32-bit MCU was eventually implemented later on. There are open-source prototyping board designs available for use with the NodeMCU firmware, which is itself an open-source project. Both the firmware and the designs for the prototyping boards are available for free online. NodeMCU is a lua-based open-source firmware that was developed specifically for Internet of Things applications. ESP-12E is the module that is based on the ESP8266 MCU and it is the module that is responsible for running this firmware. It offers Wi-Fi at 2.4 GHz and is compatible with WPA2 and WP2.

Features	Specifications
Microcontroller	Tensilica 32-bit RISC CPU Xtensa LX106
Operating voltage	3.3V
Input Voltage	7-12V

Table 1: Specifications and features of NodeMCU

B. DHT11 Sensor

DHT11 sensor is a digital sensor at low cost used to sense temperature and humidity. This sensor can interface with Arduino, raspberry and other microcontrollers such as ATmega328P, etc., to provide instantaneous measurements of temperature and humidity. DHT11 sensor available in both sensor and module version to measure both temperature and humidity by DHT11 temperature and humidity sensor.

Features	Specifications
Working Voltage	3.5V to 5.5V
Working current	0.3mA (measuring) 60uA (standby)
Working Temperature Range	0°C to 50°C
Working Humidity Range	20% to 90%

Table 2: Specifications and features of DHT11 Sensor

C. Soil Moisture Sensor

A soil moisture sensor is a sort of electrical sensor that can be obtained for a low cost and is used to measure the amount of soil moisture present. This sensor is capable of measuring the soil's total water volume. The two basic components of this sensor are the sensing probes and the sensor module. Sensing probes constitute the initial component. After allowing the current to flow through the soil, the resistance value is measured and compared to the soil's moisture content. The sensor module processes the data read from the sensor probes and then converts the processed data into a digital or analogue output. The soil moisture sensor can therefore provide both digital output (D0) and analogue output (A0).

Features	Specifications
Working Voltage	5V DC
Working current	<20mA
Working Temperature	10-30°C

Table 3: Specifications and features of Soil moisture Sensor

D. ESP32 Cam

The ESP32-CAM is a full-featured microcontroller with an integrated video camera and microSD card adapter. It is inexpensive and easy to use, making it an excellent option for Internet of Things devices that require a camera with advanced functionality such as image tracking and recognition. Using the design included in the sample software provided by Espressif, you may develop a web-based camera with a sophisticated control panel. Once you have mastered the device's programming, you will discover that it is extremely simple to use once the necessary procedures have been completed.

Features	Specifications
Working temperature	-20 °C ~ 85 °C
Storage environment	-40 °C ~ 90 °C, <90%RH
Processor	ESP32-D0WD
RAM	Internal 512KB + External 4M PSRAM
Built-in Flash	32Mbit
Wi-Fi protocol	IEEE 802.11 b/g/n/e/i
Power supply	5V

Table 4: Specifications and features of ESP32 cam

E. GPU

The graphics processing unit, sometimes known as a GPU, has rapidly become one of the most essential components of contemporary computing technology, usable in both residential and commercial settings. The graphics processing unit

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
Volume 3, Issue 1, January 2023
Impact Factor: 6.252

(GPU), which was designed for parallel processing, is used in a wide range of applications, including graphics and video rendering. Despite being most well-known for their capabilities in gaming, GPUs are rapidly being utilised in disciplines such as artistic creation and artificial intelligence (AI). GPUs were first designed to accelerate the rendering of three-dimensional graphics. They were able to increase their capacity over time by becoming increasingly programmable and adaptable. By utilising more sophisticated lighting and shadowing techniques, graphics programmers have been able to build more captivating visual effects and more realistic scenes. Other engineers began utilising the capabilities of graphics processing units to significantly accelerate additional activities in high-performance computing (HPC), deep learning, and other fields (GPUs).

3.2 Libraries

A. TensorFlow

TensorFlow can build sophisticated applications with high precision by utilising multi-layer neural networks. Image processing, video analysis, real-time object detection, decision making, audio modification, and finding anomalies in datasets are just a few of the applications that can benefit from its use. TensorFlow provides the techniques and framework required to implement machine learning using artificial neural networks (ANN) and decision trees in order to compute massive numerical datasets while maintaining accuracy. TensorFlow is an:

- **Open-source:** TensorFlow is a library that is open-source, which means that programmers have the ability to simply add more functions and make it more compatible with a variety of datasets..
- **Easy to build models:** TensorFlow gives you the flexibility to use several levels of abstraction according to your specific requirements. You may utilise the distribution approach on a variety of hardware configurations for large training projects without having to change the model.
- **Powerful experimentation for research:** Users are able to construct and train sophisticated models with the help of TensorFlow, without having to compromise on speed or performance.

B. Platforms Used

- **Thonny IDE:** A self-proclaimed independent developer by the same name created the free software development programme known as Thonny for use on personal computers. Python is one of the programming languages that can be used in conjunction with this integrated development environment (IDE), which allows for the creation of a variety of applications. The Thonny Integrated Development Environment is what's utilised to get programmes to execute on a microcontroller board that supports micro python.
- **Anaconda:** Anaconda You can write code in the computer language Python and then execute that code on the Python platform, which is a free and open-source platform. It was developed by continuum.io, a business that specialises in the Python programming language's application development. When it comes to scientific computing, data science, and machine learning, the Anaconda platform is by far the most popular method for learning and utilising Python. The Anaconda software guides you through the process of constructing an environment suitable for a wide range of Python and package version combinations. In addition to this, you may use Anaconda to instal, uninstall, and update packages in the environments of your projects. In addition, Anaconda allows you to quickly launch any desired project with just a few clicks of the mouse.

3.3 Algorithms

A. Random Forest

The Random Forest is supervised learning algorithm developed by google that is used to solve classification and regression in field of machine learning and this model is trained by the labelled data. We know that forest means great number of trees, and more trees there are, more resilient forest will be. A classifier known as Random Forest takes the average of numerous decision trees that have been applied to different subsets of a given dataset to increase the accuracy of the dataset's predicting capabilities. It is predicated based the idea of ensemble learning, which refer to practise of integrating numerous classifiers for solving a difficult problem and to enhance the functionality of the model [13]. When it comes to training our model, we make use of random forest because, in comparison to other methods, it provides a higher level of accuracy. When compared to the other unsupervised learning algorithms illustrated in figure3, Random Forest provides predictions with a higher level of accuracy.

The following are the procedures that were covered in our training for ml models:

- Step 1: The first step of the model is to select samples at random from the training data that has been provided.
- Step 2: The algorithm will build a decision tree for each piece of training data.
- Step 3: In the third step, choice tree will be averaged before voting take place.
- Step 4: At a last step, choose prediction result which received the most votes to get the final prediction result.

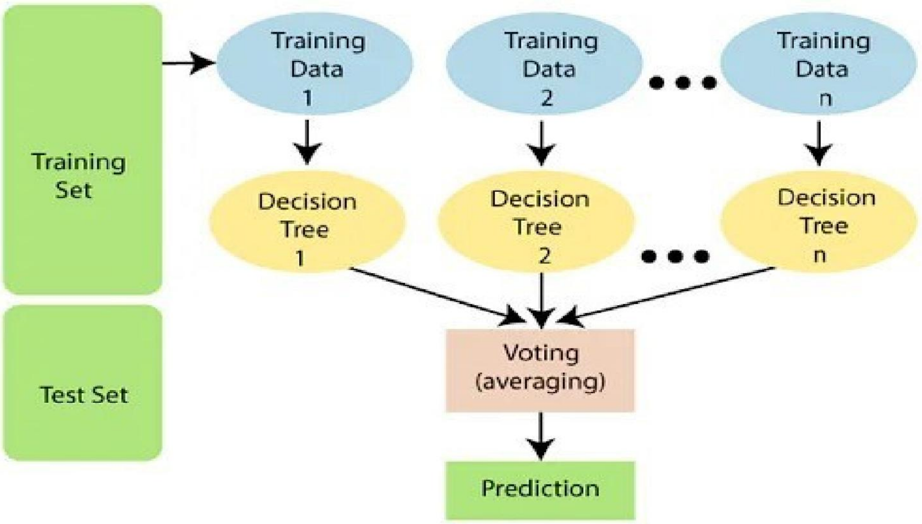

Figure 1: Random Forest

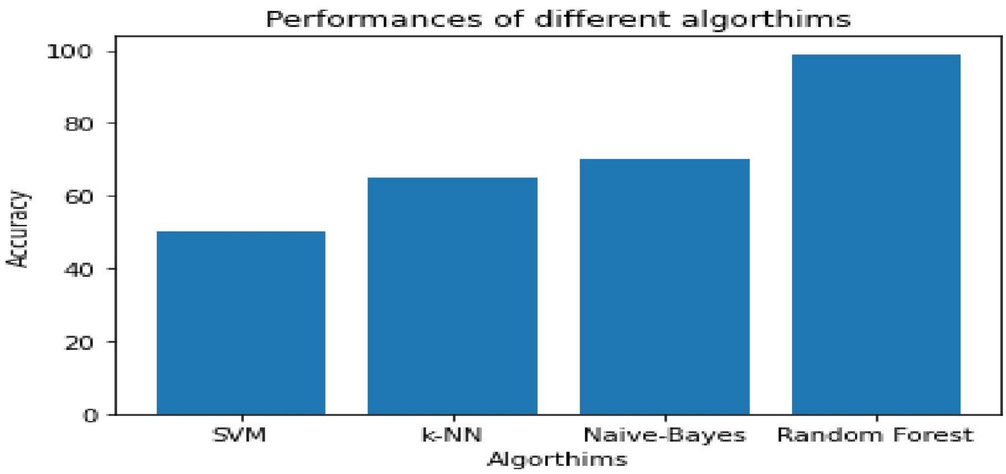

Figure 2: Performance levels of unsupervised learning algorithims

3.4 Convolutional Neural Networks (CNN)

In its full name, CNN stands for "convolutional neural network." CNNs, or "convolutional neural networks," are a type of artificial neural network (ANN) typically used in deep learning for image recognition, text recognition, object recognition, and classification. CNNs (convolutional neural networks) are used in deep learning to recognise objects within an image or text.CNN, which stands for "convolutional neural network," is a commonly employed technology in machine vision applications. Additionally known as "convnets" and "CNN," The subset of deep neural networks responsible for doing visual data analysis When attempting to identify items based on image and video data, this design style is scrutinised. It is used in applications such as video or image recognition, neural language processing (NLP), and other types of processing [14]. The CNN algorithm includes four distinct layers. Their respective names are Convolution Layer, ReLU Layer, Pooling Layer, and Fully Connected Layer.

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
Impact Factor: 6.252
Volume 3, Issue 1, January 2023

Convolution Layer:

A convolution layer has many filters capable of performing convolutional operations. The term "convolution" refers to a mathematical operation that, when applied to two functions, yields a third function that represents how the form of one function is modified by the form of another function. In a similar sense, CNN is more comparable to a convolution operation, in which we take multiple measurements but give greater weight to those that are close together. Therefore, at this time, we conduct revised measurements, the outcome of which is a weighted average of the previous measurements performed in such a way that the more recent data are given greater weight than the older ones. The filter input will be converted into a feature map. Equation: $St = \Sigma x_{t-a}\, w_{t-a} = (x*w)_t$ X = input image pixel matrix W = filter * = convolution

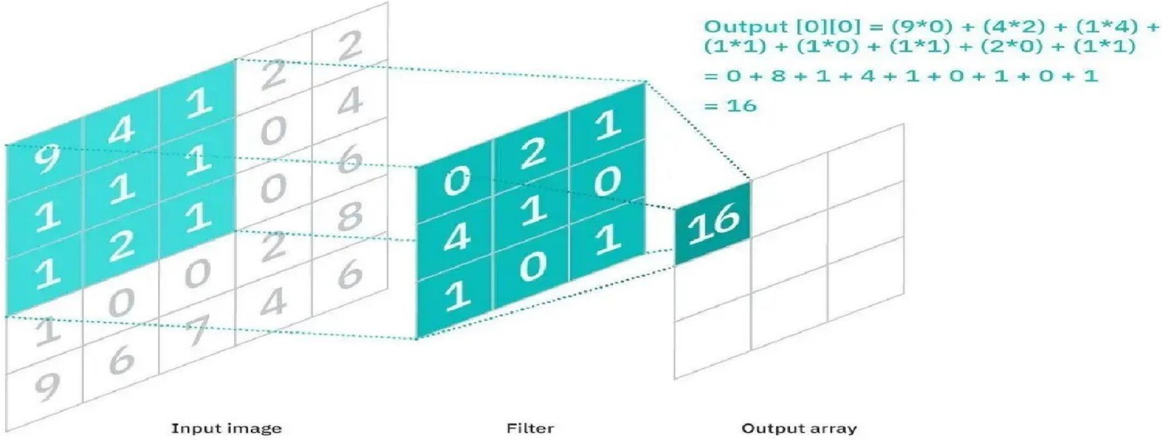

Figure 3: Convolution Layer

Relu Layer

After the feature maps have been acquired, the final step is to move them to a ReLU layer. The rectified linear unit, often known as the "ReLU," is not a discrete component of the convolutional neural network process. The convolution procedure, which was covered in the last tutorial, must be finished with an additional step. Some instructors and writers discuss both phases individually, but in our case, we will consider both phases to be components of the first phase of our technique. Some educators and authors discuss the two processes independently. If you've read the last section on artificial neural networks, you should be familiar with the rectifier function illustrated in Fig 4.

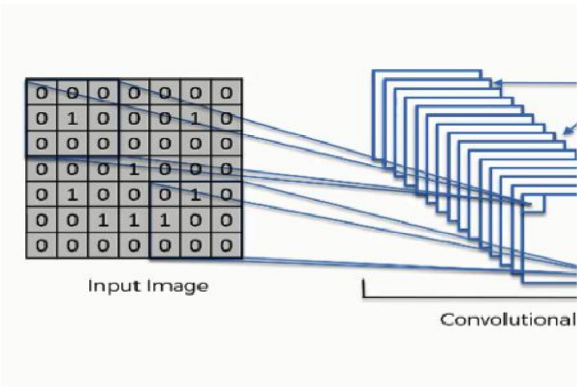

Figure 4: ReLU Layer

The application of the rectifier function will raise the level of nonlinearity in our photographs. The fact that pictures are intrinsically non-linear is the driving force behind this objective. When you examine an image closely, you will notice that it contains numerous elements that are not organised in a linear fashion. In order to compensate for any linearity that may be injected into an image as a result of the convolution technique, the rectifier serves to further distort the image's existing linearity. Observing the following image, we can observe its transformation as it undergoes the convolution operation, followed by the rectification operation. This will assist us in comprehending how the process operates.

IJARSCT

ISSN (Online) 2581-9429

International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)

Volume 3, Issue 1, January 2023

Impact Factor: 6.252

Pooling Layer

A pooling layer is now added after the rectified feature map has been processed. A procedure called "pooling," which is a sort of down sampling, can be used to lower the dimension of the feature map. The pooling layer is another one of CNN's structural components. Dimensionality reduction is the principal purpose it serves. This is one of the most effective methods for mitigating the issue of overfitting. Using a variety of filters, it may also be used to identify the edges and corners of objects. There are two distinct methods of resource pooling.

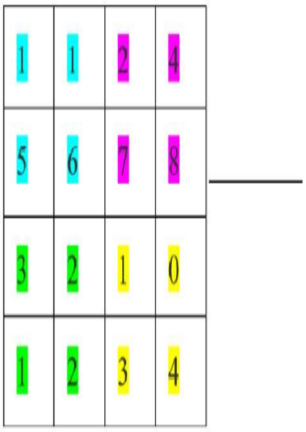

Figure 5.1: Max pooling

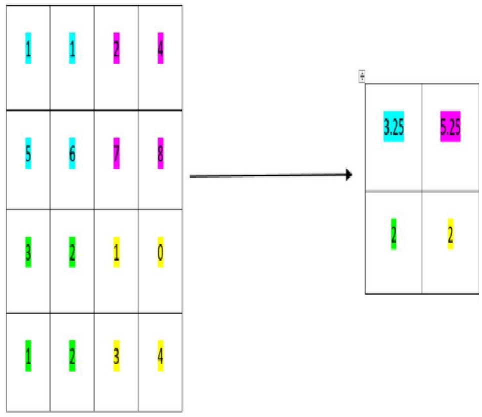

Figure 5.2: Average pooling

- **Max pooling:** Max pooling takes the value with the largest filter size in fig 5.1
- **Average Pooling:** The average value for each block of values will be determined by using the avg pooling method shown in fig 5.2

Output Layer

This consists of a layer that is fully connected, which is then followed by an activation function called SoftMax, which determines the output classes.

Fully Connected Layer

In the same way that the output of a neural network would be sent to fully connected layers, the output matrix of this layer is transformed into vector form and then sent to those layers. While neurons in the layer below are responsible for weights and bias, this layer assigns vectors to each neuron. It generates the output matrix and the last layer in order to produce the output image and its class name. Using a two-layer design, the CNN output is provided by both the SoftMax and Logistic layers. The logistic layer handles both binary and multiclass classifications, whereas the SoftMax layer handles multiclass classifications. This is CNN's general architecture, and transfer learning is utilised in the retraining of CNN's networks.

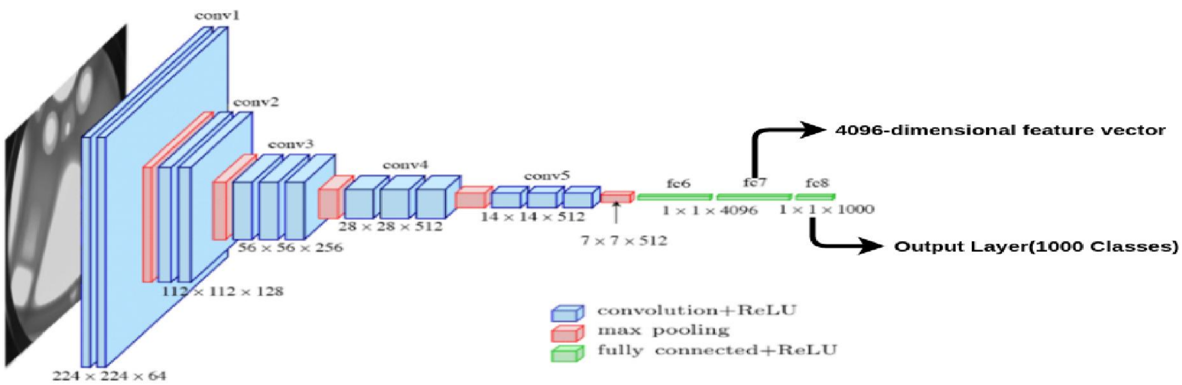

Figure 6: CNN model

IJARSCT

ISSN (Online) 2581-9429

International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)

Impact Factor: 6.252

Volume 3, Issue 1, January 2023

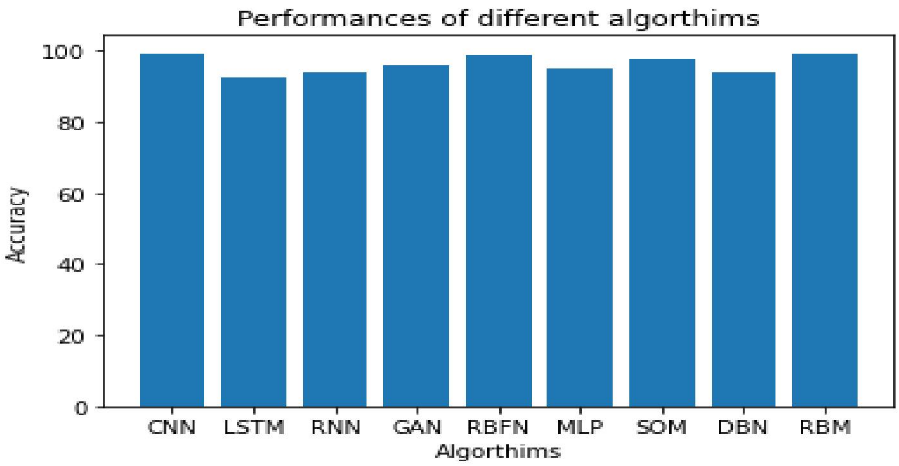

Figure 7: Performance of deep learning models

3.5 Experimental Setup

Detection of plant disease takes place in two stages.

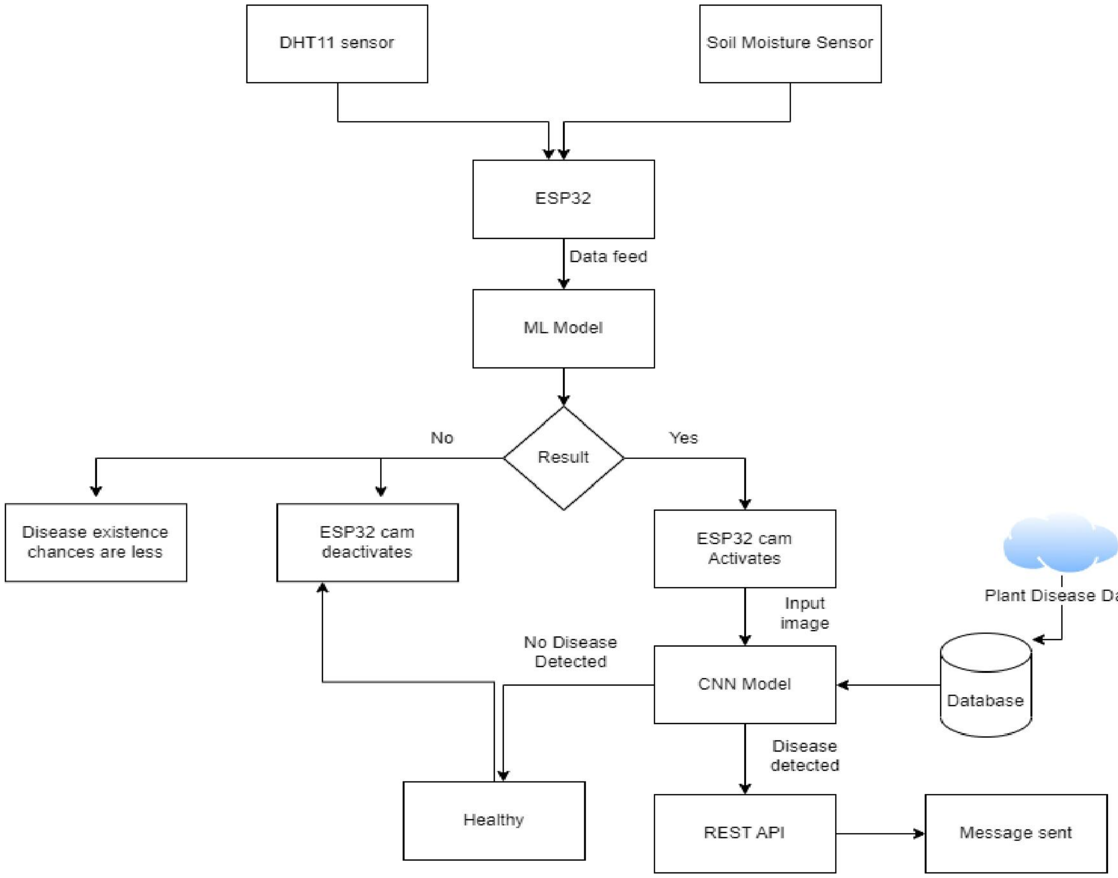

Figure 8: Flowchart

3.6 Training with ML Model

To collect continuous, real-time data from the field, the DHT11 sensor and the soil moisture sensor must be connected to the ESP32 board. These data are then fed into a machine learning model trained on a random forest. The machine learning model is trained using the Kaggle-obtained Plant Village data set. Plant Village's data gathering contains information regarding the temperature, humidity, and soil moisture levels of crops at various growth stages. These data will also be combined with the daily sensor data, resulting in an increase in the total amount of data. A machine-learning model was also provided with these data in order to make a disease detection prediction. Based on the prediction, additional data are added to the dataset, and a machine learning model is formed after each iteration of the model. If the disease is identified by the ML model, the outputs from that model and the sensor data are forwarded to the CNN model in order to produce a more precise conclusion. In the event that this does not occur, the ESP32 will enter a power-saving sleep mode until the disease is detected.

3.7 Verified by the CNN Model

If a disease is detected during the initial stage, the ESP32 camera will be enabled to photograph the plant's leaves. These images will then be submitted to a CNN model, which will use them to predict the type of illness that will develop on the plant. The CNN model is trained in Spyder IDE with 10,000 images using the TensorFlow framework and Spyder IDE. It then searches the database and the internet cloud for potential cures for the sickness and delivers the pertinent information to farmers so they can treat the ailment at an earlier stage. The remedies are delivered to the farmer via the registered mobile number, and the farmer is encouraged to place an order for the required fertilisers. In the case that it detects the presence of pesticides, it will immediately begin emitting ultrasonic waves with a frequency of more than 20,000 Hz to eliminate pests without harming humans. If the model does not predict any disease, an error message is sent to the ML model, instructing it to revisit the training dataset and update the details based on the new information. By undergoing training with two distinct models, one can achieve a more precise output. By going through these two steps, the system's ability to make accurate judgements is enhanced. In the first stage, the ML model is validated using only the sensor values from the DHT11 and soil moisture sensors. In the second stage, however, the CNN model is trained using images from the ESP32 camera in addition to the sensor readings. This two-stage approach is also used to reduce the power consumption of the ESP32 camera, which is considerably higher than that of other sensors. In order to satisfy the requirements, the cam module is currently functioning in wake-up mode; otherwise, it will transition to sleep mode.

3.8 Results and Discussion:

Category Name	Precision (%)	Recall (%)	F1-Score	Accuracy (%)
Diseased	91.07	95.41	93.29	94.71
Not Diseased	98.36	94.71	96.43	96.84
Overall	96.12	96.25	96.16	95.99

Table 4: Classification report of ML model

Two key steps comprise the process of discovering patterns using a method based on machine learning. Training the classifier by extracting weights from the training examples is the initial step of the procedure. Following this, the system analyses the correctness of the system using the test samples. Therefore, it was necessary to divide the total samples into training samples and testing samples, respectively. Following the conclusion of the training phase, the classifier will use twenty percent of the total samples to evaluate the system's data classification accuracy. The sensors transmit data to the machine-learning model constructed using the Random Forest approach. The data set passed when compared to the data set in the plant village dataset. The model can assess whether or not the plant is diseased. Every single prediction made by the machine learning model will turn out to be erroneous. Sometimes, misinterpretations within the data sets result in erroneous conclusions. Based on the trained data reported in Table 4, the accuracy of ML models is calculated to be 95.99%. Each of the 10,000 distinct training datasets includes humidity, temperature, and soil moisture readings from the crop. These characteristics are used to determine whether or not a plant is unhealthy. After training in the random forest model is complete, the ESP32 data is added to the ESP32 camera's input images. This is accomplished by awakening the camera from the sleep mode it was placed in automatically by the system in order to conserve energy. The fully connected layer is always the final layer in a neural network, regardless of the presence or absence of convolutional layers. This holds true even when both sorts of network levels are present. This is true regardless of whether a network's layers are physical or logical. Regardless of the number of convolutional layers present in the network, this remains the same. This remains true despite the fact that the network may have a variable total number of convolutional layers. This is due to the fact that the layer that has retained all of its connections has also retained all of the connections from the layers that preceded it. This is because the layer that has maintained all of its connections has also maintained all of the existing connections. This is due to the fact that both the layer and this layer have kept all of their connections [15]. This particular layer is not unique to CNNs due to the aforementioned property; rather, it may be utilised in a vast array of different neural network types in addition to CNNs. This specific layer is not unique to CNNs. This is due to the fact that CNN does not own it exclusively. This is due to the fact that CNN does not place a high value on the topic. As its input, this layer receives a vector, which is then used to generate another vector, which is then transmitted through this layer as its output. This layer is responsible for conducting the multiplication operation as part of its responsibilities. This layer is able to perform the functions of receiving and transmitting vectors since it can construct vectors at both its input and output locations. As its input, this layer receives a vector, which is then used to generate another vector, which is then transmitted through this layer as its output. This layer is responsible for conducting the multiplication operation as part of its responsibilities. This layer is able to perform the functions of receiving and transmitting vectors since it can construct vectors at both its input and output locations. To do this, the input values are first subjected to a linear combination, and then maybe an activation function is applied to them. This is done in order to attain the desired results. This course of action is taken to get the intended outcome. Only by taking this course of action would it be feasible to achieve the desired outcome. The desired outcome is achievable, but only if the exact steps indicated in this action plan are followed out in the correct sequence. If the specific procedures outlined in this action plan are carried out in the exact order, it will be feasible to achieve the desired outcome; however, this is the only condition under which this will be achievable. The last layer, which is responsible for connecting everything, generates a vector of size N, where N is the number of categories that must be allocated to the images that must be classified. This layer provides an exhaustive collection of connections [16]. Due to the fact that each node in this layer is connected to each node in each sublayer, the size of the resulting vector is N. Each component of the vector in the vector space provides an indication of the degree to which an image is likely to correspond to a particular category. This signal can be used to assist in determining whether or not an image belongs to the category. Using this information, it is possible to determine which category a photograph belongs to. Accuracy and loss during the fifty epochs while the model were being trained the number of times a model is taught is denoted by epochs. At each consecutive epoch, as illustrated in Figure 10, both accuracy and loss exhibit opposite trends. Figure 11 depicts the results of the subsequent studies conducted after the training. The overall accuracy of the model is 99.32%, while the error margin is 0.1%. Due to the potential for error in the first model, the disease is predicted twice in the model. This is done in order to assess the possibility that the disease will be accurately predicted using the two-stage prediction method. As the initial step, determine whether or not the crop is infected with a disease. In the case that a plant becomes diseased, the system examines the database to identify the specific disease and then shows what it finds. If the leaves look to be in good health, the dataset used by the ML model will be updated, resulting in more accurate outcomes. Once the disease has been discovered, the system notifies the farmer of the disease's characteristics as well as any necessary crop remedies.

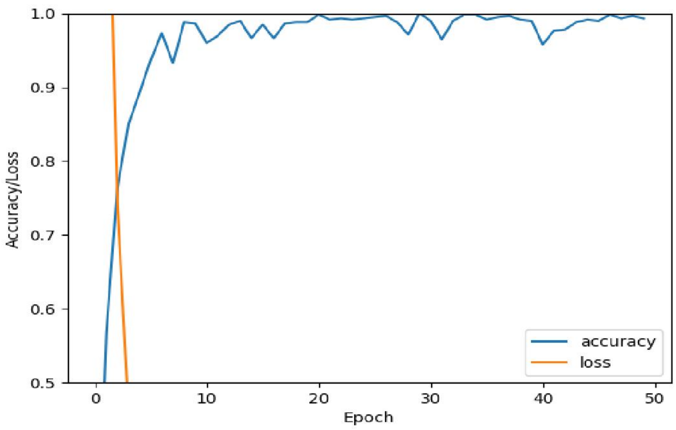

Figure 9: Accuracy/Loss vs epoch

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
Impact Factor: 6.252
Volume 3, Issue 1, January 2023

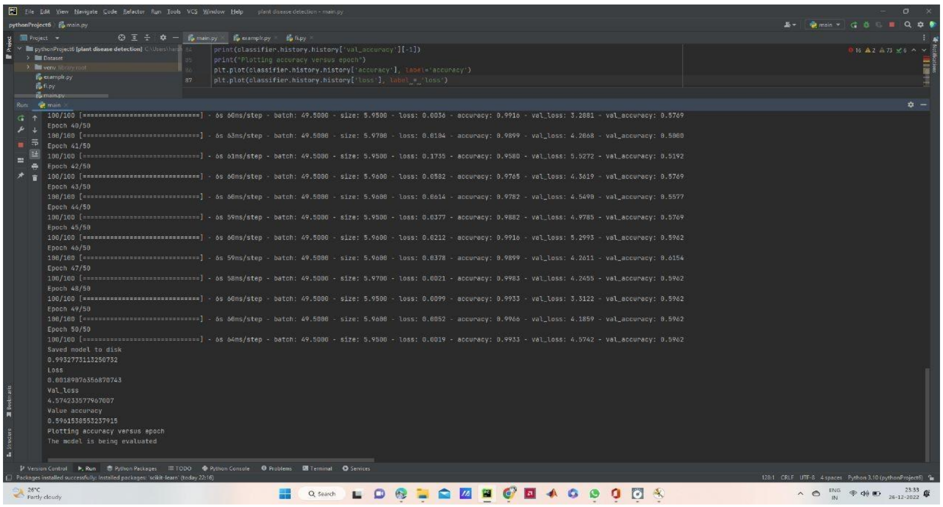

Figure 10: Experimental Results

IV. CONCLUSION

Using two models, microcontrollers and sensors, to acquire real-time data, the work reported in this paper concerns the identification of plant diseases. The first model is a machine learning model that was taught using the Random Forest method, while the second model is a deep learning model that was taught using a convolutional neural network (CNN). The accuracy of both the CNN model and the random forest model is 95.99%. The accuracy of the CNN model is 99.2%. Due to the model's ability to adapt the sensors' and system's resting mode, it can also live longer than three months. In the future, it may be possible to introduce other plant species, each with its own set of diseases and textural qualities. In addition, crop care could be automated using sensors and actuators to control the watering and fertilising operations. There is opportunity for improvement in the supplied models' predictions, which can be performed by extracting as many unique traits as possible from the plant's leaves. To acquire precise measurements of the pertinent characteristics, the model should be deployed in conjunction with a variety of sensors.

REFERENCES

[1]. Ms. Mary Vineela .PMSc (N) Lecturer C H N,Naryana College of Nursing, Nellore, Chemical composition of Tulsi, Benefits of holy basil(tulsi) leafs.

[2]. Das SK and Vasudevan DM. Tulsi: The Indian holy power plant. Natural Product Radiance. 2006; 5:279-83.

[3]. Triveni, Kuldeep Kumar, Amit Kumar Singh, Rahul Kumar, Vaishnavee Gupta, Kishu Tripathi* Smt. Vidyawati College of Pharmacy, Jhansi (U.P.) – 284121, India. Ocimum sanctum Linn: A Review on Phytopharmacology and Therapeutic Potential of Tulsi, International

[4]. Brahimi, M., Arsenovic, M., Laraba, S., Sladojevic, S., Boukhalfa, K., Moussaoui, A., 2018. Deep learning for plant diseases: Detection and saliency map visualisation, in: Human and Machine Learning. Springer, pp. 93–117.

[5]. Brahimi, M., Boukhalfa, K., Moussaoui, A., 2017. Deep learning for tomato diseases: classification and symptoms visualization. Applied Artificial Intelligence 31, 299–315.

[6]. Nageshwar Jaiswal, and Vivek Sarnaik (2021) “Detection of Plant Leaf Disease Using CNN Algorithm”. International Journal of Research in Engineering and Science (IJRES) ISSN (Online): 2320-9364, ISSN (Print): 2320-9356 www.ijres.org Volume 9 Issue 9 ǁ 2021 ǁ PP. 35-44

[7]. Ramya R, Kiran M, Marimuthu E, Naveen Kumar B, Pavithra G (2020) “Plant Monitoring and Leaf Disease Detection with Classification using Machine Learning-MATLAB”. International Journal of Engineering Research & Technology (IJERT) ISSN: 2278-0181 Published by, www.ijert.org RTICCT - 2020 Conference Proceedings.

[8]. Naveen Kishore Gattim, Subba Reddy Pallerla, Polaiah Bojja, Tallem Pavan kumar Reddy, Vempati Nikhil Chowdary, V Dhiraj, Sk Hasane Ahammad (2019) “Plant Leaf Disease Detection Using SVM Technique”.

Volume7, No.11 November 2019 International Journal of Emerging Trends in Engineering Research https://doi.org/10.30534/ijeter/2019/367112019

[9]. Vijay Singh, A.K. Misra (2016) "Detection of plant leaf diseases using image segmentation and soft computing techniques". Elsevier, Information processing in agriculture 4, issue 1.

[10]. Salwa Khalid Abdulateef, Mohanad Dawood Salman (2021) "A Comprehensive Review of Image Segmentation Techniques". Iraqi Journal for electrical and electronic engineering 17(2):166-175 DOI:10.37917/ijeee.17.2.18

[11]. K. Padmavathi, K. Thangadurai (2016) "Implementation of RGB and Grayscale Images in Plant Leaves Disease Detection-Comparative study". Indian journal of science and technology, vol 9(6), DOI: 10.17485/ijst/2016/v9i6/77739, February 2016

[12]. Shaltout, A., Spachos, P., & Gregori, S. (2016, October). Power management modelling of a photovoltaic system for a wireless sensor network. In 2016 IEEE 21st International Workshop on Computer Aided Modelling and Design of Communication Links and Networks (CAMAD) (pp. 201-206). IEEE

[13]. Hatuwal, Bijaya & Shakya, Aman & Joshi, Basanta. (2021). Plant Leaf Disease Recognition Using Random Forest, KNN, SVM and CNN. Polibits. 62. 13-19. 10.17562/PB-62-2.

[14]. Harte, Emma. (2020). Plant Disease Detection using CNN. 10.13140/RG.2.2.36485.99048.

[15]. K. Abhirami, G. Chandra Praba, R. Sugantha Lakshmi, 2021, IOT Based Paddy Crop Disease Identification and Prevention System using Deep Neural Networks and Image Processing, INTERNATIONAL JOURNAL OF ENGINEERING RESEARCH & TECHNOLOGY (IJERT) ICRADL – 2021 (Volume 09 – Issue 05)

[16]. Varshney, Divyanshu & Babukhanwala, Burhanuddin & Khan, Javed & Saxena, Deepika & Singh, Ashutosh. (2021). Machine Learning Techniques for Plant Disease Detection. 10.1109/ICOEI51242.2021.9453053.