International Journal of Engineering and Management Research
e-ISSN: 2250-0758 | p-ISSN: 2394-6962
https://ijemr.vandanapublications.com

Peer Reviewed & Refereed Journal
Volume-14, Issue-1 (February 2024)
https://doi.org/10.5281/zenodo.10689196

Plant Health Monitoring System Using Machine Learning

Anand Kumar Dohare[1] and Adnan Ahmad Khan[2]

[1]Assistant Professor, Department of Information Technology, Greater Noida Institute of Technology (Engineering Institute), Gautam Buddh Nagar, Uttar Pradesh, INDIA

[2]Scholar, Department of Information Technology, Greater Noida Institute of Technology (Engineering Institute), Gautam Buddh Nagar, Uttar Pradesh, INDIA

[1]Corresponding Author: dohare.anand@gmail.com

Received: 08-01-2024 **Revised:** 25-01-2024 **Accepted:** 19-02-2024

ABSTRACT

Agriculture has transformed into more than just a means to feed growing populations; it's a crucial sector in India, engaging over 70% of the workforce and ensuring sustenance for a vast number of people. Plants play a pivotal role in ecosystems, supporting human life and wildlife by providing food. Preserving plant health is imperative, particularly in detecting diseases, as it directly impacts the quality and quantity of agricultural yields.

This paper focuses on the technologies that are being used in plant health monitoring system which is being adopted nowadays in agriculture to make farming easy, for example image processing approaches for plant disease detection. Manually monitoring plant diseases is a difficult task. A manual plant disease monitoring system needs additional processing time and plant disease knowledge. As a result, a method for identifying plant diseases that is quick, automated, and accurate is required. As a result, image processing techniques are utilised to detect, process, and identify plant diseases since they are quick, automated, and accurate. Visualization is a traditional way of identifying diseases in plants, however it is not as effective in detecting diseases linked with plants. As a result, we can give a superior option, one that is both fast and precise, by employing image processing techniques that are more trustworthy than certain older methods.

Keywords-- CNN, Image Processing, Machine Learning, Plant Disease, Tensorflow

I. OBJECTIVE

The main objective of this research paper is to understand the emerging technologies being used nowadays in farming that is making farming an easy to do, manage and keep record of the data generated by collecting the health report of the plants. Also, to understand image processing techniques that can help farmer if implemented by them to monitor their plants and what disease their plants are having. In this research paper I have mentioned one of the image processing techniques that can detect plant disease easily by one click. After the farmer get to know about the disease, he can use pesticides that will help him to protect the plants. This research paper also discusses how technology are being adopted by farmers in farming to take farming to next level using technology and make it smart farming.

II. INTRODUCTION

Plants serve as a significant energy source but also contribute to the global warming crisis. Pathogens, both emerging and endemic, can cause substantial harm to plants, resulting in financial setbacks. The outward appearance of agricultural produce holds immense value, influencing both its selling price and consumer behavior. Therefore, ensuring quality through rigorous grading and quality control systems is pivotal in cultivating robust, high-quality plants. Diseases in plants can cause extensive economic and productivity losses in agriculture, making disease management a challenging task. Symptoms like colored spots or streaks on leaves or stems are typical signs of diseases caused by fungi, bacteria, or viruses. These symptoms are usually identified through manual inspection by trained specialists using the naked eye. However, factors like fatigue, visual acuity, work pressure, and environmental conditions like poor lighting and temperature can affect the accuracy of expert diagnosis and decision-making in detecting plant diseases.

As a result, this is an inefficient and time-consuming procedure. It may be costly, as constant expert observation on large farms.

The study of automated plant disease analysis is crucial as it holds the promise of benefiting vast crop fields by swiftly identifying disease indicators as they appear on leaves. Researchers aim to develop a speedy, cost-effective, and precise technique for disease identification by employing pixel-based statistical analysis to compute leaf area. This concept involves creating a mobile application enabling farmers to capture plant images using their smartphones. These images will undergo assessment based on visible symptoms, and advanced image processing methods will be employed to detect the type of infection.

The role of image processing techniques in the identification and diagnosis of plant diseases in their early stages is discussed in this research and what techniques are now being used to make agriculture smart and less time consuming is discussed.

III. LITERATURE REVIEW

Tanha Talaviya, Dhara Shah, Nivedita Patel, Hiteshri Yagnik, and Manan Shah explored various applications of artificial intelligence in agriculture, focusing on irrigation, weeding, and spraying facilitated by sensors embedded in robots and drones. They outlined practices that minimize water and pesticide usage, preserve soil fertility, optimize manpower, and enhance productivity and quality in agriculture.

Trimy Neha Tete and Sushma Kamlu delved into image processing techniques for plant disease detection and classification, utilizing the K-means cluster and feed-forward backpropagation algorithms. However, their study was limited by a small dataset.

Dr. S. Balaji, R. Saravanan, L. Suvetha, R. Anandhi, and M. Jeevadharshini focused on plant health monitoring through an IoT-based system employing sensors to monitor soil moisture, pH, temperature, and air quality. They proposed using microcontrollers to compare collected sensor values with stored data for result determination.

Vignesh Dhandapani, S. Remya, T. Shanthi, and R. Vidhy explored image processing techniques for plant disease detection, employing the K-means and Support Vector Machine (SVM) algorithms. They highlighted the efficacy of these algorithms, also used in medical fields like cancer detection, for plant disease identification and processing.

IV. TRADITIONAL PRACTICES

Recent developments indicate that technology is significantly changing how livestock management operates, affecting poultry farms, dairy farms, cattle ranches, and related agribusinesses. Technologies like robotics and machine learning have brought about a revolution in agriculture, impacting operations both on small and large scales. Today, technology has become indispensable for farmers, particularly in developed nations. Essentially, it encompasses the skills enabling us to create tools and machines to fulfill our needs. Some prevalent technologies utilized in farming include harvest automation, autonomous tractors, seeding and weeding, and drones. These innovations pave the way for new methods to cultivate and distribute food to larger populations. Agriculture, a sector extending beyond farming and ranching to include agribusiness, faces the challenge of supporting the current global population with limited land using outdated technology. The emergence of new technologies has the potential to transform agriculture in the foreseeable future, addressing these challenges and revolutionizing the agricultural landscape.

4.1 Using Sensor Technology

The adoption of precision agriculture technologies is gaining popularity among farmers, as it becomes increasingly affordable to deploy sensors strategically across farmlands. These sensors, combined with image recognition technologies, empower farmers to remotely monitor their crops from anywhere in the world. The real- time traceability provided by sensors offers insights into current conditions of farms, forests, or water resources. This technology proves invaluable in monitoring and managing livestock, optimizing crop production, and promoting environmental sustainability by conserving water, minimizing erosion, and reducing fertilizer runoff into local rivers and lakes. Additionally, RFID sensors play a role in tracking food products from the field to the store after harvest, ensuring a transparent and accountable supply chain.

4.2 Precision Agriculture Technology

Precision agriculture, also known as precision farming, focuses on implementing more precise and efficient farming methods for planting and crop cultivation. This approach integrates information and communication technology (ICT) with optimal agricultural practices, potentially increasing agricultural output while minimizing environmental harm. Its impact on food production is particularly crucial amid a growing global population.

Precision agriculture relies on various technologies like drones, IoT, GPS guidance, sensors, robotics, autonomous vehicles, and telematics to enable its practices.

4.3 Machine Learning

Machine learning (ML), a branch of artificial intelligence (AI), finds applications in the agricultural sector. It involves automatically identifying significant patterns within datasets. In contemporary agriculture, there's a focus on conserving water, optimizing nutrient and energy usage, and adapting to changing climates. ML in agriculture enhances disease diagnosis accuracy and aids in predicting crop diseases. Through AI and ML algorithms, farmers can extract valuable insights from data, boosting efficiency, productivity, and yields. Additionally, ML algorithms have relevance in the manufacturing facets of agriculture.

4.4 Smart Farming

Smart farming utilizes IoT solutions to gather real-time data on various elements like weather, soil

Peer Reviewed & Refereed Journal
Volume-14, Issue-1 (February 2024)
https://doi.org/10.5281/zenodo.10689196

conditions, crop growth stages, equipment status, and labor expenses. This approach involves predictive analytics to enable more informed decision-making in agriculture. It encompasses the adoption of information and communications technologies (ICT) to automate and enhance agricultural processes, essentially a facet of precision agriculture. Smart farming is versatile, applicable to small-scale family farming, complex agricultural systems, and organic farming. Its emergence replaces outdated and unreliable traditional farming methods, improving the efficiency of farming practices. Through wireless sensor networks, it continuously monitors soil properties and environmental factors, while employing smart irrigation systems to dispense necessary nutrients based on crop requirements.

4.5 Automation Technology

This encompasses any tool designed to alleviate the workload of operators, integrating sensors, computers, feeding mechanisms, and robots. Robotics, by eliminating human involvement, receives considerable attention. Automation stands as the primary focus driving technological advancements in agriculture, already implemented worldwide. Cost-effective GPS-guided tractors and planters allow drivers to oversee systems primarily. The intelligent, self-directed tractor holds immense potential to revolutionize various aspects of the agricultural sector. Automation technology has the capacity to enhance safety and productivity in farming. While most field crop operations are mechanized across production, handling, and storage, labor remains an unavoidable element in agriculture, regardless of the farm's degree of automation.

4.6 Drones

Drones, whether autonomous or remotely controlled, are versatile aerial vehicles operated by aerodynamic forces, functioning without a human onboard. They're emerging as innovative tools in agriculture. To gain an aerial perspective of their fields, farmers rely on planes or drones. Drones find application in crop monitoring and chemical spraying, offering the capability of producing 3D imaging for soil quality prediction. Equipped with sensors, drones facilitate monitoring crop and soil health or detecting weed data within crops.

4.7 Using AI & IOT

Observation systems based on Artificial Intelligence (AI) and the Internet of Things (IoT) are in high demand, offering precise data extraction and analysis. These systems monitor the impact of physical conditions like humidity, temperature, soil conditions, and light intensity on plant growth through IoT- based monitoring. Various sensors such as DHT11, LDR, DS18B20, soil moisture sensors, Noir cameras, single-board microcontrollers, and Application Programming Interfaces (APIs) collect data essential for understanding plant growth. Further analysis of these parameters is carried out using diverse Machine Learning (ML) algorithms.

Algorithms like Provision Regression, Gradient Boosting Classifier, and Linear Support Vector Classifier (SVC) prove effective in analyzing physical factors influencing plant growth. The agricultural sector faces challenges amid the growing global population. AI-based technological solutions have enabled farmers to enhance output, improve production quality, and expedite the time taken to bring crops to market. AI facilitates automation, risk reduction, and simplification of farming tasks for increased efficiency.

Emerging technologies aid in optimal crop selection and improve hybrid seed options suitable for farmers' requirements by assessing how seeds respond to varying weather and soil conditions. This data collection minimizes the risk of plant diseases. AI-powered chatbots and machine learning algorithms enable tailored communication and understanding of natural language, benefiting agriculture by offering tailored answers to farmers' queries, guidance, and recommendations.

V. PROPOSED METHODOLOGY

This section focuses on detecting disease on plant leaves using Python, so for this we are using image processing technique. The primary goal of this planned study is to assist farmers who have suffered losses owing to insufficient knowledge about a variety of diseases. In this research paper we have developed a model by which we can detect disease in plants by using image processing. We have used Jupyter notebook for implementing our codes and techniques. To detect the disease, numerous image processing techniques are used. Image classification is used to get useful descriptions that may be used in subsequent processes. With image processing, Convolutional Neural Network (CNN) is also used, CNN is a powerful algorithm for image processing. This algorithm is one of the best algorithms for the automated processing of images. Many companies use these algorithms to do things like identifying the objects in an image. And so, as we have also used CNN algorithm to train our data.

We have first imported the libraries to develop CNN model that will help us to detect disease in the plant. So, libraries we have used are keras, matplotlib. Using those libraries, we can do some of the important things like first very important to perform augmentation configuration to get highest accuracy performance basically augmentation configuration is the process in which create different images by rescaling, rotating, performing width and height shift, zoom & flip images. Below we will be discussing about the two libraries more briefly.

Python: Python is a high-level computer language for general-purpose programming that is interpreted. Its fantastic libraries and tools aid in the effective completion of image processing tasks. OpenCV must be installed in Python. 'Open-source computer vision library' founded in 1999 by a group of enthusiastic coders to integrate Image Processing into a wide range of programming languages. It runs on Windows, Linux, Android, and Mac with C++, C, and Python interfaces. It is one of the libraries used in Python for image processing. It finds and identifies the Leaf and illnesses on the Python Web Framework using the Leaf Identification technique.

5.1 System Architecture

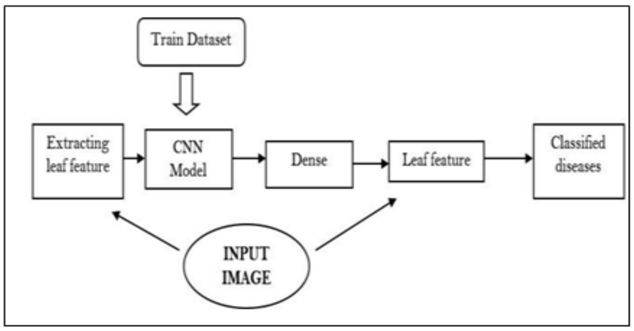

Figure 5.1

1. Keras

Keras is a simple framework for creating deep learning models that are based on TensorFlow. It is intended to be used quickly to create deep learning models. Keras, on the other hand, is a fantastic solution for deep learning models. Keras simplifies and improves the performance of high-level neural network APIs by utilising numerous optimization methodologies. It is capable of the following:

- API that is consistent, simple, and extensible.
- Simple structure - Without any frills, it is straightforward to get the desired result.
- It works with a wide range of platforms and backends.
- It's an easy-to-use framework that works on both the CPU and the GPU.
- Extremely scalable computation.

2. Matplotlib

Matplotlib is a fantastic Python visualisation package for creating two-dimensional array charts. It is a multi- platform visual data tool based on numpy arrays and designed to function with the whole Scipy stack. One of the most major benefits of visualisation is that it gives us visual access to enormous amounts of data in comprehensible formats. Matplotlib has a number of plot types such as line, bar, scatter, histogram, and so on.

3. Tensorflow

Tensorflow is a library that is open source for numerical computation and large-scale machine learning. It combines machine learning and deep learning models together for analysis. Tensorflow provides various levels of abstraction, allowing us to select the best one for our purposes. Tensorflow provides you with freedom and control with tools such as the Keras functional API for creating complicated topologies. Using the high-level Keras API, you can build and train models. It also provides a robust ecosystem of add-on libraries and models to explore with. Tensorflow enables developers to design data flow graphs, which are structures that represent the movement of data via a graph or set of processing nodes. Tensorflow offers several functionalities to programmers via the Python language. It distributes the nodes of a dataflow graph across multiple computers in a cluster as well as among various processing units within a machine, such as multicore CPUs, general-purpose GPUs, and custom- designed ASICs known as Tensor Processing Units (TPUs). Tensorflow is used for a wide range of tasks, with a focus on deep neural network training and inference. Tensorflow is used in production by a number of Google services.

4. Proposed Model

The Convolutional Neural Network (CNN) stands out as an artificial neural network that has gained significant traction across various computer vision tasks. It bridges the gap between traditional feed-forward neural networks and adaptive filters, presenting an intriguing approach to adaptive image processing. CNNs are designed to autonomously learn spatial feature hierarchies via backpropagation, employing distinct building blocks like convolution layers, pooling layers, and fully connected layers.

These networks utilize one or more layers of two-dimensional filters combined with non-linear activation functions and down-sampling, creating a framework tailored for geographically or temporally scattered input processing. CNNs effectively limit network weights and connections, showcasing attributes like translation invariance and spatially local connections, known as receptive fields. The "weight-sharing" feature in CNNs curtails the number of free parameters.

Despite their utilization in character recognition, the full potential of CNNs remains to be fully realized. Nonetheless, their ability to automatically learn and adapt spatial hierarchies has made them pivotal in the realm of computer vision tasks.

5.2 Proposed Model Functions

```
In [198]:
#Building CNN model
cnn_model = keras.models.Sequential([
                    keras.layers.Conv2D(filters=32, kernel_size=3, input_shape=[150, 150, 3]),
                    keras.layers.MaxPooling2D(pool_size=(2,2)),
                    keras.layers.Conv2D(filters=64, kernel_size=3),
                    keras.layers.MaxPooling2D(pool_size=(2,2)),
                    keras.layers.Conv2D(filters=128, kernel_size=3),
                    keras.layers.MaxPooling2D(pool_size=(2,2)),
                    keras.layers.Conv2D(filters=256, kernel_size=3),
                    keras.layers.MaxPooling2D(pool_size=(2,2)),

                    keras.layers.Dropout(0.5),
                    keras.layers.Flatten(), # neural network building
                    keras.layers.Dense(units=128, activation='relu'), # input layers
                    keras.layers.Dropout(0.1),
                    keras.layers.Dense(units=256, activation='relu'),
                    keras.layers.Dropout(0.25),
                    keras.layers.Dense(units=4, activation='softmax') # output layer
])
```

Figure 5.2

Conv2D: It is a two-dimensional convolution layer that generates a tensor of outputs by wrapping a convolution kernel with the layer's input. In image processing, a kernel is a convolution matrix or filter that may be used for blurring, sharpening, embossing, edge detection, and other effects by executing a convolution between a kernel and an image.

MaxPooling2D: This function is used to maximise the value from the specified size matrix, and it is repeated for the following two levels.

Flatten: This method is often used to reduce the dimensions of a convolved image.

Dense: This is the regular densely linked neural network layer. It is the most popular and widely used layer. This is the hidden layer that is utilised to build a fully connected model.

Dropout: This function is used to avoid overfitting on a dataset. It is a sort of regularisation that aims to reduce overfitting by boosting testing accuracy at the price of training accuracy. Dropout is used to reduce overfitting by selectively modifying the network configuration during training.

Image Data Generator: It resizes the image and zooms in and out the image given and flips it horizontally. This Image Data Generator includes all image orientations.

Training Process: To prepare data from the train dataset directory, use the function train datagen, flow from directory. The target size specifies the image's goal size. To generate test data for the model, test datagen and flow from directory are used, and the others are the same as before. To fit the data into the model, the fit generator is utilized; additional criteria used include steps per epochs, which tell us how many times the model will run for the training data.

Epochs: This method defines how many times the model will be trained in both forward and backward passes. Validation process: As part of the validation process, it is used to provide validation and test data into the model. The validation phases show the number of validation and test dataset samples.

5.3 Training & Testing Model

The dataset undergoes preprocessing, involving tasks such as reshaping, resizing images, and converting them into an array format. Similar preprocessing steps are applied to test images. The dataset encompasses 38 distinct plant leaf diseases, allowing any image to be used for software testing. Model training involves utilizing a Convolutional Neural Network (CNN) with layers like Dense, Dropout, Activation, Flatten, Convolution2D, and MaxPooling2D. The CNN learns from the training dataset to recognize diseases in test images. Upon successful training, the software is capable of predicting diseases if the plant species is present in the dataset. Following successful training and preprocessing, the software compares the test image with the trained model to predict the disease.

5.4 Implementation and Analysis

	disease	count_images
0	Corn_(maize)___Cercospora_leaf_spot Gray_leaf_...	411
1	Corn_(maize)___Common_rust_	954
2	Corn_(maize)___healthy	930
3	Corn_(maize)___Northern_Leaf_Blight	788
4	Potato___Early_blight	800
5	Potato___healthy	122
6	Potato___Late_blight	800
7	Tomato___Bacterial_spot	1702
8	Tomato___Early_blight	800
9	Tomato___healthy	1273
10	Tomato___Late_blight	1528
11	Tomato___Leaf_Mold	762
12	Tomato___Septoria_leaf_spot	1417
13	Tomato___Spider_mites Two-spotted_spider_mite	1341
14	Tomato___Target_Spot	1124
15	Tomato___Tomato_mosaic_virus	299
16	Tomato___Tomato_Yellow_Leaf_Curl_Virus	4286

Figure 5.4.1

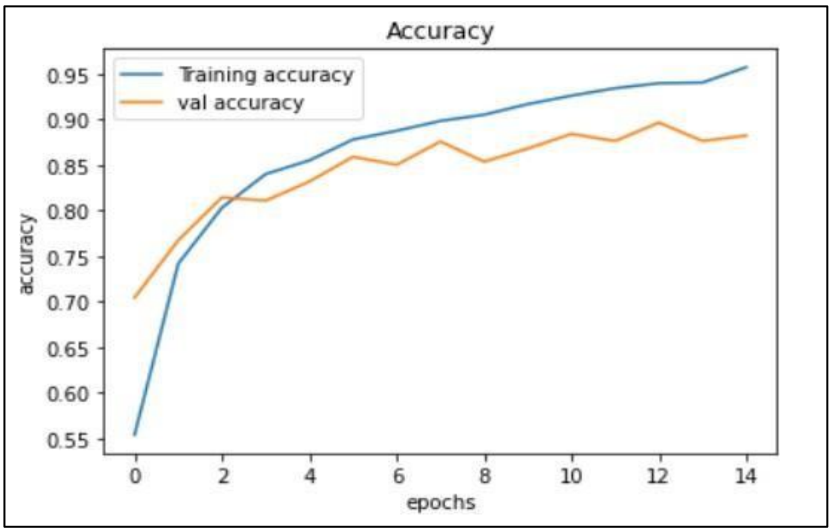

Figure 5.4.2

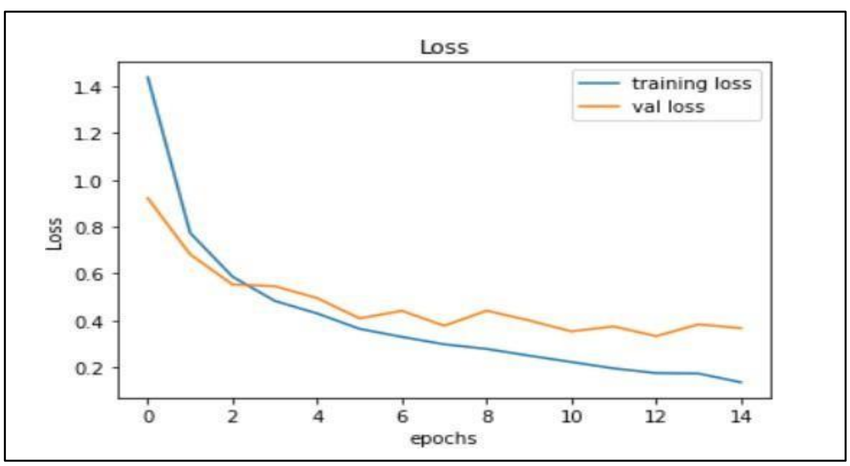

Figure 5.4.3

Peer Reviewed & Refereed Journal
Volume-14, Issue-1 (February 2024)
https://doi.org/10.5281/zenodo.10689196

The above image shows the graph representation of our model accuracy and loss. We can see in the graph that our model training accuracy is more than 90%. As with every epoch our model accuracy is increasing, and loss is decreasing.

If we give a new image to the model to predict its outcome. The model predicts any one of the classes from the 17-class mentioned before.

5.5 Model Deployment using Android Studio

TensorFlow has made deployment to mobile much easier. TensorFlow Mobile is also about to become deprecated according to Google. So, in this section you will come to know about the deployment of the model using android studio java.

So, following are the steps to deploy your tensorflow model using android studio: -

Step 1. Change the Extension of the Model

After your model has been created have to change the extension of the model to .tflite in order to use the model into your application.

Step 2. Add Tflite Model to your Project

You have to first create a new project in the android studio, and you will add your tflite model to your project.

Step 3. Copy the Tflite Code

You have to copy the code from tflite file and paste it on your MainActivity.java file inside the function from where the model will work after giving the image of the plant.

Step 4. Test the App & Resolve the Bugs if Exist

So, after designing your app and creating a nice interface, you have to test the application whether it is working fine or not and check for the bugs and resolve if found any.

Figure 5.5.1

Figure 5.5.2

VI. CONCLUSION

The process of leveraging convolutional neural networks (CNN) for disease detection in plants involves several essential steps. Beginning with the dataset, an array of pre-processing steps such as reshaping, resizing, and converting images into a standardized array format is undertaken. These transformations ensure uniformity and compatibility across the dataset, making it amenable for the subsequent analysis. Similarly, the test image undergoes these same pre-processing steps to align with the training data, facilitating effective comparison and analysis. This dataset encompasses a diverse array of approximately 38 distinct plant leaf diseases, serving as a comprehensive repository for potential test samples within the software. The training phase is pivotal as it empowers the CNN model to discern and categorize these various diseases accurately. Through this process, the model learns to associate specific visual patterns and characteristics with particular diseases, enabling it to make informed predictions based on visual cues within the images. The architecture of the CNN model involves a series of interconnected layers. These layers, including Dense, Dropout, Activation, Flatten, Convolution2D, and MaxPooling2D, collaborate in tandem to extract and process critical features from the images. The convolution layers focus on identifying distinct patterns within the images, while pooling layers facilitate the extraction of dominant features. The Dense layers contribute to the classification process, utilizing the learned features to make informed decisions regarding disease identification. Once the CNN model completes its training phase successfully, the software gains the capability to proficiently identify diseases within the plant species included in the dataset. Leveraging the learned patterns and associations, the model processes the test image, comparing it against the trained model for disease prediction. This approach enables the software to accurately predict diseases based on the visual characteristics exhibited in the images. Through this sophisticated integration of CNN architecture and comprehensive datasets, the software can effectively contribute to plant disease diagnosis and subsequent agricultural management decisions, aiding in crop protection and yield optimization.

REFERENCES

[1] Rahat Yasir, Md & Ashiqur Rahman. (2014). Dermatolomical disease detection usinm imame processinm and artificial neural netgork. *ICECE, International Conference on IEEE*, pp. 687-690.

[2] Samajpati, thavini J & Sheshang D. Degadwala. (2016). Hybrid approach for apple fruit diseases detection and classification usinm random forest classifier. *Communication and Signal Processing (ICCSP), International Conference IEEE*, pp. 1015-1019.

[3] Ashwini S.R, Chethan, Shivashankar, B.R. Harish, R. Karthik & Karan D Bafna. (2018) Wireless sensor network for environmental radiation monitoring using IOT. Presented at: *Third IEEE International conference on recent trends in electronic, information and communication technology*.

[4] Mr. Ashish Nage Prof. Ram Meghe Institute of Technology & Research, Badnera, Prof. V.R. Raut Prof. Ram Meghe Institute of Technology & Research, Badnera. (2019). detection and identification of plant leaf diseases based on python. *International Journal of Engineering Research & Technology, 8*(5).

[5] Dr. S. Balajia, R. Saravanana, L. Suvethab, R. Anandhib & M. Jeevadharshinib. A survey on plant health monitoring using IoT. Received: 10 January 2021; Revised: 12 February 2021; Accepted: 27 March 2021; Published online: 28 April 2021.

[6] Trimi Neha Tete Department of Electrical & Electronics tirla Institute of Technology Mesra Ranchi, Jharkhand) Sushma Kamlu Department of Electrical & Electronics tirla Institute of Technology Mesra Ranchi, Jharkhand. plant disease detection using different algorithms. *Second International Conference on Research in Intelligent and Computing in Engineering*. DOI: 10.15439/2017R24 ACSIS, Vol. 10 ISSN 2300-5963.

[7] Vignesh Dhandapani, S. Remya, T & Shanthi, R. Vidhy Department of Computer Science Engineering,VSB Engineering College, Karur, Tamil Nadu, INDIA. (2018). Plant health monitoring using image processing. *International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)*, 5(3).

[8] Nishanth Vijay Supervisor: Joe Steinhauer Examiner: Tove Helldin. (2021). *Detection of plant diseases in tomato leaves*. Master Degree Project in Informatics with a Specialisation in Data Science Spring term.

[9] S. Bharath, K. Vishal Kumar, R. Pavithran & T. Malathi. (2020). Detection of plant leaf diseases using CNN. *International Research Journal of Engineering and Technology (IRJET)*, 7(5).

[10] Sreeram Sadasivam, Vishwanath, Vadhri & Supradha Ramesh. (2015). *Smart plant monitoring system.* July 2015 DOI:10.13140/RG.2.1.2915.6563/1.

[11] R. Sui & J. Baggard. (2015). Wireless sensor network for monitoring soil moisture and weather conditions. *Applied Engineering in Agriculture 31*(2), 193-200.

[12] Tanha Talaviya, Dhara Shah, Nivedita Patel, Hiteshri Yagnik & Manan Shah. (2020). *Artificial intelligence in agriculture*. The Authors. Publishing services by Elsevier B.V. on behalf of KeAi Communications Co. Ltd. This is an open access article under the CC BY license.