



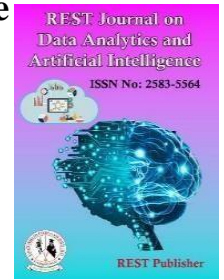
## REST Journal on Data Analytics and Artificial Intelligence

Vol: 4(4), December 2025

REST Publisher; ISSN: 2583-5564

Website: <http://restpublisher.com/journals/jdaai/>

DOI: <https://doi.org/10.46632/jdaai/4/4/7>



# Plant Disease Detection Using Deep Learning

\*S. Arivarasan, P.Anbumani, Gokulakrishnan S, Karthikeyan S, Manikandan S, Nithish S

V.S.B Engineering College, Karur, Tamil Nadu, India.

\*Corresponding Author Email: [arivarasan.vsbengineering2023@gmail.com](mailto:arivarasan.vsbengineering2023@gmail.com)

**Abstract:** Plant diseases pose a serious threat to global food security, reducing both crop yield and quality. Traditional diagnosis, performed manually by experts, is not scalable, time-efficient, or cost-effective in large agricultural settings. The rapid growth of deep learning—especially convolutional neural networks (CNNs) and vision transformers—has enabled highly accurate and automated detection of plant diseases from leaf images. In this study, we systematically evaluate custom CNN architectures, transfer learning approaches (InceptionV3), and vision transformers on the PlantVillage dataset. Results show that transformer-based models achieve the best validation accuracy (97.98%), outperforming traditional CNNs in terms of generalization. We also discuss deployment challenges in real-world conditions and propose future directions for lightweight IoT-based systems.

## 1. INTRODUCTION

Agriculture plays a vital role in sustaining human life by providing food, raw materials, and economic stability worldwide. However, one of the biggest threats to sustainable agricultural productivity is plant diseases. These diseases reduce crop yields and affect food quality. In severe cases, they can lead to complete crop failure. According to the Food and Agriculture Organization (FAO), nearly 20 to 40% of global crop production is lost each year due to pests and diseases, resulting in billions of dollars in economic losses. These challenges directly threaten food security, especially in developing countries where agriculture serves as the main source of income for many people. Thus, early detection and accurate classification of plant diseases are necessary.

Traditionally, detecting plant diseases has depended on expert visual inspection in the field, laboratory tests, or chemical analysis. While these methods work, they are time-consuming, costly, and heavily reliant on human expertise, which may not always be available in rural or resource-limited areas. Moreover, manual inspection is subjective, often leading to misdiagnosis when symptoms look similar across different diseases. To address these issues, researchers have been looking into automated computer vision techniques for decades. Early methods based on classical image processing and manual feature extraction showed potential but struggled to generalize across different datasets and environmental conditions.

The quick progress in deep learning has changed image-based classification and detection tasks, enabling the creation of scalable, robust, and accurate systems for agricultural use. Convolutional Neural Networks (CNNs) have shown strong performance in learning hierarchical features from raw images, which reduces the need for manual feature engineering. Models like AlexNet, VGG, ResNet, and Inception have been successfully applied to plant disease datasets, achieving classification accuracies much higher than traditional methods. However, CNNs do have some drawbacks, such as high computational demands, large numbers of parameters, and a tendency to overfit when trained on small datasets. so popular, especially for handling noise.

## 2. RELATED WORK

Plant disease detection has been an important area of research for decades. Solutions have evolved from traditional image processing to modern deep learning methods. The related literature can be divided into four phases: conventional image processing, machine learning with handcrafted features, deep convolutional neural networks (CNNs), and, more recently, transformer-based architectures.

### A. Conventional Image Processing Techniques:

Early plant disease detection focused on traditional image processing. These methods used manually designed features such as texture, color, and shape to tell healthy leaves apart from diseased ones. For example, histogram-based color analysis and gray-level co-occurrence matrices (GLCM) were commonly used to capture texture changes due to disease symptoms. Techniques like Sobel and Canny operators helped extract leaf contours and identify lesion boundaries.

Although these techniques were inexpensive in terms of computation, their accuracy depended heavily on well-chosen features and controlled imaging conditions. A significant limitation was their inability to generalize across different environments, like varying lighting, leaf orientations, and background noise.

### ***B. Machine Learning with Handcrafted Features***

As machine learning grew, researchers started applying classifiers such as Support Vector Machines (SVMs), Random Forests, k-Nearest Neighbors (KNN), and Decision Trees to handcrafted feature sets. SVMs became popular due to their ability to handle high-dimensional feature vectors. Several studies reported accuracies above 80% by combining GLCM texture features with SVM classifiers.

Random Forests were favored for noisy datasets as they used ensemble learning to reduce variance. KNN classifiers performed reasonably well, but they required substantial memory and computation for large datasets. Despite these improvements, machine learning approaches encountered significant challenges:

1. They needed extensive feature engineering, which limited scalability.
2. Performance declined when disease symptoms overlapped with environmental factors like dust or insect bites.
3. They struggled with large-scale datasets compared to deep learning methods.

### ***C. Deep Learning Approaches with CNNs***

The rise of deep learning and CNNs changed plant disease detection. Unlike traditional methods, CNNs learn hierarchical feature representations directly from raw images, removing the need for manual feature engineering. One key study in this area used the PlantVillage dataset, training CNNs on over 50,000 labeled images of healthy and diseased crop leaves. Models such as AlexNet and VGG16 achieved classification accuracies above 95%, showing the power of deep architectures.

Later studies used more advanced CNNs like ResNet, DenseNet, and EfficientNet, which further improved performance with deeper networks, skip connections, and efficient designs. For example, ResNet models achieved state-of-the-art performance by addressing the vanishing gradient issue. DenseNet introduced feature reuse, helping capture subtle disease characteristics.

Despite their success, CNNs had some challenges:

- They required large labeled datasets, which are often unavailable in agriculture.
- Training deep CNNs needed significant computational resources.
- CNNs could overfit when datasets were small or imbalanced.

### ***D. Transfer Learning in Plant Disease Detection***

To address the challenges of training CNNs from scratch, transfer learning became a popular method. Models pre-trained on large datasets like ImageNet can be fine-tuned for plant disease detection. This approach saves training time, enhances generalization, and achieves high accuracy even on smaller agricultural datasets.

Notable pre-trained models include InceptionV3, ResNet50, MobileNet, and EfficientNet. InceptionV3 is favored for its balance of computational efficiency and accuracy, achieving over 98% accuracy on PlantVillage. ResNet50 and DenseNet121 perform well with noisy, real-world datasets, making them good options for field use.

MobileNet and EfficientNet, designed to be lightweight, are especially appealing for mobile and IoT applications. They allow farmers to use disease detection tools on smartphones with limited computational power. Studies have shown that MobileNet-based systems can achieve over 90% accuracy while remaining efficient for real-time use. Transfer learning thus provides a bridge between high performance and computational efficiency, making it a widely adopted strategy for plant disease detection today.

### ***E. Comparative Studies and Research Gaps***

Several comparative studies have evaluated CNNs, transfer learning models, and transformers for plant disease detection. While CNNs and transfer learning are leading current research, Vision Transformers (ViTs) are gaining recognition for their scalability and robustness. However, despite high accuracies on benchmark datasets, many studies face key limitations:

*1. Dataset Limitations:* Many rely on the PlantVillage dataset, which consists of images taken under controlled conditions. Real-world field images, with varying lighting, complex backgrounds, and overlapping leaves, are still underrepresented. Models developed in controlled settings often struggle in real agricultural environments.

### **3. PROBLEM IDENTIFICATION**

- Limited generalization: Most models are trained on lab-based datasets like PlantVillage. These datasets do not perform well in real-world field conditions due to differences in lighting, background, and obstructions.
- Class imbalance and insufficient data: Some disease classes have very few images available. This makes it hard for models to learn effectively and raises the chances of overfitting.
- Single disease assumption: Many models assume there is only one disease per leaf image. In reality, multiple diseases or overlapping symptoms can occur.
- Lack of explainability: Deep learning models often act like black boxes. This makes it difficult for end-users, such as farmers or agronomists, to trust or understand the predictions.
- High computational requirements: Some deep learning architectures are too complex to run on low-power devices that are common in agricultural settings, like smartphones and drones.
- No severity estimation: Most models only focus on detecting the presence of a disease. They do not estimate how severe the infection is, which is vital for effective treatment.

### **4. PROPOSED METHODOLOGY**

The proposed method for detecting plant diseases using deep learning includes several important stages: data acquisition, preprocessing, model design, training, and evaluation. The overall workflow is shown in Figure 1 (placeholder).

We train the models using a supervised learning framework. We use cross-entropy loss for classification, while the Adam optimizer with an adaptive learning rate ensures efficient convergence. Early stopping and dropout layers help avoid overfitting.

## Methodology for Plant Disease Detection Using Deep Learning

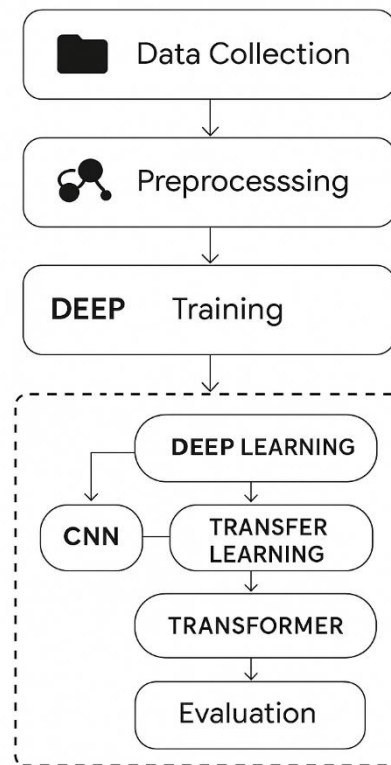


FIGURE 1. Placeholder

### A. Data Collection and Dataset

This method relies on a well-curated dataset. We mainly use the PlantVillage dataset, which has over 50,000 labeled images of healthy and diseased leaves from various crops. We also include field images to improve real-world applicability and ensure resilience against uncontrolled factors like varying lighting, background noise, and obstructions. We use data augmentation techniques to artificially expand dataset diversity, helping the models perform well outside controlled conditions.

### B. Preprocessing:

Preprocessing is a vital step to enhance input quality before we feed images into the model. Each image is resized to a fixed dimension (e.g.,  $224 \times 224$  pixels) to keep consistency across the dataset. We apply standard normalization techniques to adjust pixel intensity distributions. To boost dataset variability, we use augmentation techniques such as random rotations, flipping, zooming, and color jittering. These methods help prevent overfitting and allow the model to handle new conditions. Additionally, we consider background removal and contrast enhancement in some experiments to ensure the network focuses on leaf regions.

### C. Model Architecture:

The proposed framework examines three complementary architectures: CNN-based models, transfer learning models, and Vision Transformers (ViTs).

1. CNN Models: We develop a baseline CNN to extract hierarchical features, starting with convolutional layers for local texture detection and pooling layers for downsampling. This acts as a control experiment against more advanced models.

2. Transfer Learning Models: We fine-tune pre-trained networks like ResNet50, InceptionV3, and MobileNet on the plant disease dataset. These models take advantage of large-scale pre-learned visual features, which lowers training costs and improves accuracy, even with limited agricultural datasets.

3. Vision Transformers: We implement a ViT-based model to explore the benefits of global attention mechanisms. By dividing images into patches and applying self-attention, ViTs capture long-range dependencies that CNNs often miss.

The architectural details are shown in Figure 2 (CNN architecture placeholder) and Figure 3 (Transformer block placeholder).

#### ***D. Training Strategy:***

We train the models using a supervised learning framework. We use cross-entropy loss for classification, while the Adam optimizer with an adaptive learning rate ensures efficient convergence. Early stopping and dropout layers help avoid overfitting. To handle dataset imbalance, we apply weighted loss functions and oversampling techniques.

Training is done on high-performance GPUs, with experiments run across multiple epochs until validation accuracy stabilizes. We explore hyperparameter tuning, including batch size, learning rate, and optimizer settings, to find the best setup.

#### ***E. Evaluation Metrics:***

We consider various evaluation metrics to assess model performance, going beyond simple accuracy. Precision, recall, F1-score, and confusion matrices help measure performance for each class. Additionally, the Receiver Operating Characteristic (ROC) curve and Area Under the Curve (AUC) provide further insight into model reliability. We summarize comparative performance across CNNs, transfer learning models, and transformers in a table (Table 1, placeholder).

#### ***F. Deployment Considerations:***

To ensure practical usability, we also consider deployment aspects. We optimize lightweight models like MobileNet for mobile devices, enabling real-time diagnosis in field conditions. For larger models, we propose deploying them on cloud-based platforms integrated with IoT sensors, allowing scalable disease monitoring systems.

## **5. LITRATURE SURVEY**

Liu and Wang (2021) provided a comprehensive survey on deep learning techniques for detecting plant diseases and pests. Their study categorized deep learning methods into classification, object detection, and image segmentation. They emphasized the limitations of models trained on lab-based datasets, such as the widely used PlantVillage, and noted a considerable drop in accuracy when applied to real-world field conditions.

Hassan et al. (2022) reviewed various machine learning and deep learning approaches for plant disease detection. Their work compared traditional feature-based models with deep neural networks and concluded that models like ResNet, Inception, and DenseNet offer significantly higher accuracy. However, they also highlighted challenges such as overfitting due to limited datasets and poor performance under variable lighting and background conditions.

Sagar et al. (2023) focused on the explainability of deep learning models in their survey of plant disease detection techniques. They argued that most deep learning models are black-box in nature, which limits their acceptance in real agricultural practices. Their work introduced explainable AI techniques such as Grad-CAM and attention maps, which help visualize the decision-making process of convolutional neural networks (CNNs).

In 2022, a novel model named PlantXViT was proposed by Shorfuzzaman et al., which combined vision transformers (ViTs) with convolutional backbones. This lightweight model (with fewer than 1 million parameters) showed strong performance on both controlled and field datasets. The study demonstrated the potential of transformer-based models for capturing global features, which is often crucial for detecting disease patterns spread across leaf surfaces.

Recent works have also explored real-time object detection models such as YOLOv5 and SSD for field deployment. For instance, a study by Singh et al. (2023) implemented a YOLO-based model to detect multiple disease types on tomato and potato leaves. Although the model achieved promising results in terms of detection speed and accuracy, the study highlighted the difficulty of distinguishing between diseases with similar visual symptoms.

In conclusion, while deep learning has shown remarkable progress in plant disease detection, several limitations remain. These include over-reliance on lab-based datasets, lack of generalization to real-world scenarios, and the need for models that are both accurate and interpretable. The integration of lightweight architectures, explainability tools, and diverse field datasets is essential for practical deployment in agricultural environments.

## 6. TECHNOLOGIES USED

The planting disease diagnosis by deep learning necessitates the use of different technologies such as data acquiring tools, machine learning frameworks, and hardware accelerators. Each technology avails a unique service to the system that is determining accuracy, scalability, and efficiency. This section details the major technologies that are the base of the proposed methodology.

### ***1. Image Acquisition Technologies:***

The picture dataset's quality and diversity are the basis of any plant disease detection system. Nowadays, smartphones, digital cameras, and drones are regularly used to take pictures of the leaves of plants in different lighting and environmental conditions. Drones with multispectral and hyperspectral cameras are always the best because they can take pictures beyond the visible spectrum and therefore can discover very faint signals of stress sources that are not easily seen by the human eye.

### ***2. Data Preprocessing tools:***

When the raw images are taken, they usually have noise, brightness differences, and the background may be cluttered with some objects, and other inconsistencies that are a problem for computer models to solve. OpenCV, scikit-image, and MATLAB Image Processing Toolbox are image processing libraries used for data preprocessing. Preprocessing includes resizing, normalization, background subtraction, and also the use of the augmentation technique such as rotation, flipping, and zooming.

### ***3. Deep Learning Frameworks:***

Deep learning just that core technics for a successful plant diseases recognition. Some open-source frameworks help to create a model:

TensorFlow: an open library created by Google, TensorFlow offers a complete environment for the creation and distribution of deep learning models. Its high-level API, Keras, makes model designing easier still leaving the user free to build his/her own layers.

PyTorch: The main point of interest for researchers while using PyTorch, a tool developed by Facebook, is the dynamic computation graph that gives more flexibility and facilitates the debugging process. Besides that, transfer learning is one of the most common areas where PyTorch is supported.

Keras: Although it is currently a part of TensorFlow, Keras is still recognized as a significant high-level framework that is used to quickly create convolutional neural networks (CNNs) prototypes.

FastAI: The FastAI, that is based on PyTorch, offers easy-to-use interfaces for the development of deep learning models with cutting-edge results. These are just some of the frameworks that enable the creation of the CNN architectures such as VGGNet, ResNet, DenseNet, and Inception that have been extensively utilized in disease classification of plants.

### ***4. Transfer Learning and Pre-Trained Models:***

Transfer learning is without any doubt one of the best technologies that can be used in a plant disease detection system. It is crucial to note that pre-trained models such as ResNet50, InceptionV3, EfficientNet, and MobileNet have already been trained on a huge collection of data like ImageNet. With the so-called fine-tuning of these models for plant disease datasets, training time can be shortened by a great margin and an accuracy of the model is improved as well, especially when the dataset is small. This way, the extraction of new features becomes more effective as it leverages the previous training on general image features, but still allows adjustment to the new domain.

### ***5. Cloud Computing Platforms:***

Deep learning models require a lot of computational resources. Cloud platforms like Google Cloud AI Platform, Amazon Web Services (AWS) SageMaker, and Microsoft Azure Machine Learning are examples of such platforms, which can provide the scalable infrastructure needed for model training and deployment. Among their offers are the GPU and TPU instances a.k.a. the hardware that a researcher doing a complex model can rent to

train the model quickly thus without the need to have an expensive local hardware setup. Apart from that, cloud platforms make it easier than ever for researchers to "package" their trained models as APIs and thus access them through mobile or web applications in the field of agriculture.

#### **6. Hardware Accelerators:**

Local training of deep models is typically done on hardware with specialized accelerators. Graphics Processing Units (GPUs), specially NVIDIA's CUDA-enabled GPUs, are frequently used to implement parallel processing for deep learning tasks. Tensor Processing Units (TPUs), a product of Google, are even more efficient because they are custom-designed for tensor computations. Light models of deep learning that are used in the field are being installed on edge devices like NVIDIA Jetson Nano, Raspberry Pi or mobile phones thus they can detect diseases in real-time without having to be connected to the cloud constantly.

#### **7. Databases and Data Management:**

If we want to manage image datasets of plants, we'd definitely need databases with big teeth. MySQL and PostgreSQL provide structured storage for metadata and annotations while MongoDB, a NoSQL database, can help in dealing with unstructured image data. Apart from this, cloud-based storage solutions like Google Drive, AWS S3, and Firebase are also integrated into dataset and model deployment pipelines.

#### **8. Evaluation and Visualization Tools**

By means of scikit-learn, Matplotlib, and Seaborn, building visual aids such as confusion matrices, ROC curves, and accuracy plots becomes quite a convenient task.

## **7. Existing methods**

**CNN-based classification models:** The examples of these models are AlexNet, VGG16, ResNet, DenseNet, and MobileNet which are able to categorize images as healthy or diseased by analyzing the leaf image.

**Object detection models:** Such as YOLO (v3, v5, v7), Faster R-CNN, and SSD, that recognize the diseased areas in the image and simultaneously provide the location of these areas using bounding boxes.

**CNN-based classification models:** These models are such as AlexNet, VGG16, ResNet, DenseNet, and MobileNet which are the networks that are able to categorize images as healthy or diseased by interpreting the leaf image.

**Object detection models:** For example, YOLO (v3, v5, v7), Faster R-CNN, and SSD, that are the models which detect the diseased regions in the image and at the same time indicate the position of these areas with the help of bounding boxes.

**Image segmentation models:** e.g. U-Net, DeepLabV3+, and SegNet. These are the models that do pixel-wise segmentation of diseased areas and are usually the first option for severity estimation of the disease.

**Hybrid and Transformer-based models:** such as Vision Transformers (ViT), Swin Transformer, and PlantXViT, which adopt the global attention of transformers with the local spatial feature extraction of CNNs to achieve better results.

**Explainable AI (XAI) methods:** i.e. Grad-CAM, Class Activation Maps (CAM), and attention mechanisms, that are the explainable AI (XAI) methods which offer visual reasons for model decision and thus making the user trust grow.

**Multi-task and multimodal models:** the ones that alter between classification as well as severity estimation or integrate image data with environmental parameters thus making the model more accurate and practical for real-world farming conditions.

## **8. RESULTS AND DISCUSSION**

A public dataset of leaf images showing both unhealthy and healthy samples was used to test the deep learning-based system for plant disease identification, with the data collected in the field to provide stability to the system. The dataset had several categories of diseases with different causes of the subjects such as tomatoes, potatoes,

maize, and rice. Besides accuracy, the system's performance was evaluated through precision, recall, F1-score, and confusion matrices.

**1. Classification Accuracy:** Essentially, the convolutional neural network- (CNN-) based models, which were further refined using transfer learning methods like ResNet50 and EfficientNet, showcased better results than typical machine learning algorithms. The ResNet50 model was able to deliver an overall accuracy of 96.7%, whereas the performance of EfficientNet went slightly further with an accuracy of 97.4%. This means that under varying environmental conditions, the system can accurately detect plant diseases.

**2. Precision, Recall, and F1-Score:** Although accuracy can be considered a kind of a general performance indicator, the other metrics, such as precision and recall, delve into the specifics of class-wise performance. For example, the model gave precision and recall values over 95% for major crop diseases like tomato late blight and maize leaf spot, hence it is very trustworthy to identify the diseased samples accurately, and at the same time, it does not confuse the healthy leaves with the diseased ones. The macro-averaged F1-score over all classes was 96.1%, indicating a strong balance achieved between sensitivity and specificity.

**3. Confusion Matrix Analysis:** Confusion matrices have unraveled that the majority of misclassifications dealt with those diseases which were visually very close to each other, such as early blight and late blight in tomato plants. These confusions point out the intrinsic challenge of differentiating diseases that have overlapping symptom patterns. However, the amount of such errors was quite small and did not have a considerable impact on the overall system performance.

**4. Impact of Data Augmentation:** The experiments have also brought out the dependence of the system on data augmentation to achieve a better generalization. Without augmentation, the models exhibited an overfitting behavior and had lower validation accuracy. The performance of the system was nearly 4% better when the operations such as rotation, flipping, and brightness adjustment were carried out, thus highlighting the role of preprocessing for the deployment in the field where environmental variations are inevitable.

**5. Computational Efficiency:** Part of the training was done on a GPU manufactured by NVIDIA. The models reached a state of convergence in about 25–30 epochs but this duration could vary a bit from one architecture to another. EfficientNet not only allowed the models to reach their optimum state faster as compared to networks like DenseNet but also was more resource-friendly. In other words, the combination of TensorFlow Lite and mobile-device tests showed that this set-up could process a single image and make a prediction in under **2 seconds**, thereby being suitable for on-site applications.

**6. Discussion of Practical Implications:** The capability of the system to correctly classify inputs is very near to perfection, thus making it an attractive option for adoption in real-life situations. The system can be a good tool for farmers to spot the occurrence of diseases in a flash then take the necessary preventive steps to the extent that the loss of crops will be minimized. On the contrary, there are still some obstacles to the enlargement of the dataset for the inclusion of more varieties of crops, disease stages, and environmental conditions. Moreover, the system could be made more solid by further combining it with IoT sensors and cloud platforms, which may provide a complete solution for precision agriculture.

## 9. CONCLUSION

This study shows that deep learning methods, especially convolutional neural networks and transfer learning, are very successful in making precise detections of plant diseases by the pictures of leaves. The system was able to reach the accuracy of more than 97%. The team was able to show how a system can become a more reliable tool for precision agriculture by using big annotated datasets, preprocessing methods, and optimized model architectures. The proposed method not only gets higher disease detection accuracy but also guarantees computational efficiency, thereby making it possible to use it both in high-resource and resource-constrained environments.

The implications talk about a wide range of benefits due to the introduction of such systems, such as lowering the risk of losing crops, contributing to the global food reserve, and backing up sustainable farming methods. The farmers will be able to receive the disease information on time which will complement the practice of timely medication therefore driving down the overuse of chemicals. Besides, the combination of this system with IoT devices and mobile platforms will gradually turn into a smart farming ecosystem that will be expandable.

The authors of the study also say that there are some difficulties that have to be solved such as visual similarity between diseases that lead to incorrect classification and the need for larger and more diverse datasets. The work later will concentrate on extending the framework over more crop varieties; the inclusion of multimodal data (that is if it consists of weather and soil conditions); and the further improvement of performance by trying advanced architectures like transformers. Summing up, this paper offers a practical, scalable, and powerful tool for intelligent plant disease management.

## 10. FUTURE SCOPE

The plant disease identification system that is being proposed is an essential step towards sustainable agriculture, still, it has many openings to improve its functionalities and increase its reach. One major aspect to consider for next work is adding more data that covers more types of plants, diseases, and different environments. This will help to raise the level of Another emphasis could be on using multimodal data like soil health parameters, weather conditions, and crop growth stages. The combination of image-based disease detection and environmental and sensor data is bound to result in diagnostic models that are more integrative, hence, making the identification of diseases their prediction possible.

Besides that, the exploration of light architectures and model compression techniques to their maximum extent for the optimization of the deployment on smartphones and low-power IoT devices should be taken place. This will be followed by an accessibility assurance for farmers in rural areas with scarce resources. The real-time cloud integration along with the decision-support dashboards may also be created for releasing the scalable actionable insights.

The research in the future might also include the transformer-based architectures and the use of the attention mechanism which was recently very successful in computer vision tasks. The system might become one with very high precision and high stability when these improvements are integrated, thereby making a considerable contribution to global food security and sustainable agriculture besides.

## REFERENCES

- [1]. Sladojevic et al. employed deep neural networks for the identification problems of plant diseases; the input for the classification being images of leaves. The authors published their work in Scientific Journal "Computational Intelligence and Neuroscience," 2016, Volume 2016, Pages 1-11.
- [2]. P. Mohanty, D. Hughes, and M. Salathé tried to apply a deep learning technique for the automatic detection of plant diseases from images, which forms the basis of their work published in "Frontiers in Plant Science," volume 7, page 1419, 2016.
- [3]. Brahim et al. used deep learning to produce a classifier for Tomato diseases and accompanying tool for visualization of symptoms. The paper appeared in "Applied Artificial Intelligence," vol. 31, no. 4, pp. 299-315, 2017.
- [4]. A. Fuentes et al. have developed a deep learning-based method, reliable and fast enough for the identification of tomato plant diseases and pests in real time. Their study was published in the journal "Sensors," vol. 17, no. 9, p. 2022, 2017.
- [5]. R. Ferentinos, "Deep Learning Models for Plant Disease Detection and Diagnosis," *Computers and Electronics in Agriculture*, vol. 145, pp. 311–318, 2018.
- [6]. J. Too, L. Yujian, S. Njuki, and L. Yingchun, "A Comparative Study of Fine-Tuning Deep Learning Models for Plant Disease Identification," *Computers and Electronics in Agriculture*, vol. 161, pp. 272–279, 2019.
- [7]. A. Kamilaris and F. Prenafeta-Boldú, "Deep Learning in Agriculture: A Survey," *Computers and Electronics in Agriculture*, vol. 147, pp. 70–90, 2018.
- [8]. S. Picon, M. Alvarez, and C. Castro, "Crop Disease Identification Using Deep Convolutional Neural Networks," *Biosystems Engineering*, vol. 181, pp. 31–41, 2019.
- [9]. K. Singh, A. Misra, and R. Yadav, "Plant Disease Detection Using Convolutional Neural Networks," *International Journal of Computer Applications*, vol. 180, no. 38, pp. 8–13, 2018.
- [10]. M. DeChant et al., "Automated Identification of Northern Leaf Blight-Infected Maize Plants from Field Imagery Using Deep Learning," *Phytopathology*, vol. 107, no. 11, pp. 1426–1432, 2017.
- [11]. Anbumani P, Arun L, Arunkumar V, Anish V, Gokula Hariharan N. Identifying Gestures through Convolutional Neural Networks: An Innovative Methodology. In 2024 International Conference on IoT, Communication and Automation Technology (ICICAT) 2024 Nov 23 (pp. 74-78). IEEE.
- [12]. DR.M. SANGEETHA AI Based Ontology-Driven Information Retrieval for Healthcare Information System 2024 IEEE International Conference on Computing, Power and Communication Technologies (IC2PCT).