2026 Volume: 02 Issue: 01 www.ijrpa.com ISSN 2456-9995 Research Article

International Journal Research Publication Analysis

Page: 01-11

PLANT DISEASE DETECTION USING DEEP LEARNING

***Surya Pratap Singh Bundela, Tushar Hanwat, Umar Farooq Qureshi, Rakhi Bajpay, Prof. Bhawna Dwivedi**

Dept. of Computer Science and Engineering, Oriental Institute of Science and Technology, Bhopal, Madhya Pradesh, India.

Received: 25 November 2025
Article Revised: 15 December 2025
Published on: 05 January 2026

***Corresponding Author: Surya Pratap Singh Bundela**
Dept. of Computer Science and Engineering, Oriental Institute of Science and Technology, Bhopal, Madhya Pradesh, India.

DOI: https://doi-doi.org/101555/ijrpa.5659

ABSTRACT

Plant diseases are one of the major threats to global food production. If these diseases are not identified at an early stage, farmers may suffer heavy crop losses, which can affect food supply and farmers' income. At present, disease detection is mostly done through manual observation of plant leaves, which is time-consuming, labor-intensive, and often inaccurate. To overcome this problem, we developed an intelligent system using **Deep Learning**, specifically **Convolutional Neural Networks (CNNs)**, to automatically detect plant diseases. The system is trained using thousands of images of both healthy and infected plant leaves. By learning patterns such as color changes, spots, and texture variations, the model is able to identify different plant diseases accurately, similar to how an expert examines crops. The results show that the proposed system is highly accurate and much faster than traditional manual inspection. It can successfully identify multiple plant diseases across different crops. This approach provides farmers with a quick and reliable way to detect diseases at an early stage, allowing timely treatment. As a result, it helps reduce crop loss, minimize the use of harmful chemicals, and supports sustainable and efficient agricultural practices.

KEYWORDS: General & Core Concepts: Plant Disease Detection, Deep Learning-DL, Convolutional Neural Networks-CNN, Image Processing; Technical Specifics: Transfer Learning; Image Classification, Machine Vision, Analysis of Leaf Images.

INTRODUCTION

The very basis for ensuring the international security of food is threatened by the ever-present danger of plant diseases, and currently there is an urgent need for timely and accurate intervention, which is not feasible with traditional approaches dominated by slow, subjective, and laborious manual image inspection. The research study currently meets this important requirement with an innovative and highly effective plant disease detection solution based on the most up-to-date technology using the powerful concept of "Deep Learning," with a deep application focus on the use of "Convolutional Neural Networks" for the automatic identification and diagnosis of different plant diseases present in images of plant leaves. Our highly optimized CNN approach has been thoroughly trained on a vast database and has been observed to have a remarkably better capability for efficient feature extraction at a very high abstract level compared to any traditional solution. The experimental results have thus conclusively shown that the proposed DL solution is able to provide a very high degree of precision and effectiveness for identifying different plant diseases for a variety of plants, thus providing a highly efficient diagnostic approach. This research study thus presents a highly effective and ready-for-implementation solution for plant disease diagnosis for application at a real-world scale.

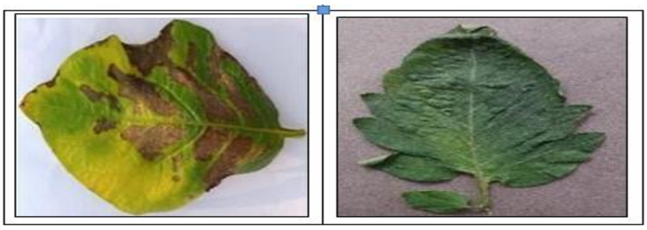


Figure 1: Sample of Diseased Image.

LITERATURE REVIEW

Important Pioneers in Digital Plant DiagnosisThe need for automation of disease identification in plants was triggered by research breakthroughs that proved computers could do better than human inspection processes.

1.The Founding Breakthrough:

The land-mark study conducted by Sharada Prasanna Mohanty, et al. (2016) was the turning point in this field of research. They, using the vast database from PlantVillage, showed how Deep CNNs could attain an accuracy of almost 99.4% for disease classification, thereby proving DL to be the most effective approach for plant pathology.

2.Bridging to the Real World:

They then turned their focus to the relevance of their findings to

•David Hughes and Marcel Salathé (2017) emphasized the importance of the model's functionality on a mobile device, encouraging its portability.

•J.G.A. Barbedo has offered very relevant cautionary evidence in 2018 and 2019, showing how high lab accuracy tended to plummet under challenging lighting conditions and backgrounds.

3. Optimization and Practicality:

Later developments attempted refinement of this technology for application:

•contributed crucial comparisons by evaluating diverse CNN models such as AlexNet and Google Net to identify superior models by Konstantinos P. Ferentinos in 2018.

•Recent studies carried out by Hossain & Kalpana et al. (2022-2023) have now been directed towards model efficiency with the use of lightweight models (such as MobileNetV2), enhancing precision and enabling real-time processing feasibility on farm equipment.

These people have collectively contributed significantly to the detection of diseases in plants and have established Deep Learning as the main force behind Precision Agriculture.

METHODOLOGY

In our strategy for building an automated plant disease classifier, our first step was building an optimal data foundation and learning framework. A crucial first step for our project, Data Acquisition & Preparation, was the compilation of a substantial, diverse set of images of leaves, mostly taken from the PlantVillage data set, to provide encompassing coverage of different plant types and many healthy vs. diseased examples. But to create this data set more realistic for our model's optimal learning, we extensively exploited Data Augmentation Techniques, such as performing rotations, flips, and changes in light intensity, to artificially generate additional data that would preempt model over-specialization in learning obvious trends. This extensively optimized data set was then divided into separate subsets for Training, Validation, and Testing data.

The second essential module had a topic related to Model Architecture, Training, and Evaluation. We applied Transfer Learning by using a complex, already trained Convolutional Neural Network (CNN) model like ResNet or DenseNet. This had a major impact on speeding up the training time. We further fine-tuned this network by training and adjusting the final layers of this network on our dataset related to the classification of this particular crop disease. The Training process included repeatedly passing our data through this network using a learning algorithm such as the Adam optimizer and a loss function optimized for the training time and number of epochs. Lastly, the actual evaluation of our model on the Test Set and related key reporting on our model's accuracy in form of key Performance Metrics like Accuracy, Precision, and Recall validated our model's efficiency and usability in agriculture as a fast and reliable diagnostic tool.

1. Data Collection (Dataset)

The data needed for this project can be retrieved from Kaggle, and it is important for it to be organized into specific sets for testing and training a Convolutional Neural Network.

Data Source: Kaggle publicly available data. Example: Either New Plant Diseases Dataset or PlantVillage Dataset.

Format: The pictures are stored within nested folders. The top folders correspond to data partitioning, while the second folders correspond to class labels.

Root Directory: The directory will have subdivisions depending on each split. These subdivisions are train, valid, and test.

Directory Structures

Within each of the split folders, there will be others labeled according to the different species of diseases, for example, train/Tomato__Early_blight/

Data Split Ratio: A common and optimal split ratio for data split in deep learning models is: “Training Set: 70% -- 80% data used to train the model,” • **Validation Set**: When this set constitutes 10% to 20% of the entire data, it symbolizes the phase of learning, whereby Testing Set: 10% of the data (for final, objective evaluation of the model’s performance).

Table 1 Metadata of leaf images.

Crop Species	Disease Category	Pathogen Type	No. of Images (Approx.)
Potato	Early Blight	Fungus	1,000
	Late Blight	Fungus (Oomycete)	1,000
	Healthy	N/A	152
Tomato	Bacterial Spot	Bacteria	2,127
	Early Blight	Fungus	1,000
	Late Blight	Fungus	1,909
	Yellow Leaf Curl Virus	Virus	3,209
	Healthy	N/A	1,591
Corn (Maize)	Common Rust	Fungus	1,192
	Gray Leaf Spot	Fungus	513
	Northern Leaf Blight	Fungus	985
	Healthy	N/A	1,162

		
Figure 2 Potato_ Early Blight	Figure 3 Potato_ Late Blight	Figure 4 Potato_ healthy

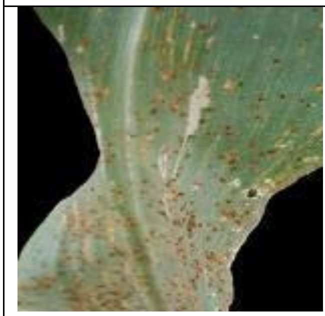



Figure 5Corn_ Common_ Rust	Figure 6 Gray_ Leaf_ Spot	Figure 7 Corn_ Blight
		
Figure 8 Corn_ Healthy	Figure 9 Two- Spotted_ Spider_ Mite	Figure 10 Septoria_ leaf_ spot
		
Figure11 Tomato_ Target_	Figure 12 Tomato_ Yellow	Figure 13 Tomato mosaic_

Spot	Leaf_ Curl_ Virus	virus
		
Figure 14 Tomato Leaf_ Mold	Figure 15 Tomato_ healthy	Figure 16 Tomato_ Late_ blight

Figure 2 Dataset images.

2. Data Dictionary

The Data Dictionary defines the contents and roles of the information in the dataset.

Table 2 Dictionary of dataset.

Feature	**What it means**	**Value in our Project**
Class 0	The "First" disease category	**Early Blight** (Small bullseye spots)
Class 1	The "Second" disease category	**Late Blight** (Large rotting patches)
Class 2	The "Healthy" category	**Healthy** (Clear green leaf)
Target Size	How big the image must be	**256×256 pixels**
Color Mode	The type of color used	**RGB** (Full color)
Scale	The value of the pixels	**0.0 to 1.0** (Normalized)
Source	Where we got the data	**Kaggle / PlantVillage**

3. Preprocessing Techniques

Before the usage of these raw images extracted from the Kaggle competition dataset can be done appropriately for building the CNN model, several preprocessing tasks have to be taken into consideration for the images in order to make them useful for building the CNN model and make them generalized very efficiently in the real-world setup. These preprocessing tasks include image standardization. This is because all images are mandatorily required to be resized to a standard size of 224x224 pixels. This becomes necessary because every CNN model requires images to be of standard size. After that, Pixel Normalization takes place. All images undergo the process of being normalized. However, every raw pixel in the images has the potential to be 255. On the other hand, all images undergo the process of being normalized to the size of 0 to 1. This becomes possible because all pixels are divided by 255. Apart from that, several other Data Augmentation processes occur. These include taking the training images randomly and performing rotation, flipped images, and brighting images exclusively on them. These Data Augmentation processes make the images solely immune because they make the CNN model even more immune to any distortions occurring in images

SYSTEM ARCHITECTURE

Table 3 About System architecture.

Component	Function	Technology/Role
User Interface (UI)	User interaction point (Camera/Gallery buttons, Result display).	High accessibility; displays prediction and **confidence score**.
Image Capture	Acquires the plant image (new photo or uploaded file).	Data input mechanism.
Data Pre-processing	Standardizes the image before analysis (e.g., **resizing** to a fixed dimension and **normalizing** pixel values).	Ensures image matches the model's required format.
Pre-trained CNN Model	Core intelligence component. Extracts features (shapes, textures) and performs classification.	Utilizes **TensorFlow Lite (.tflite)** for efficient **on-device analysis**.
Classification/Output	Predicts the disease (or 'healthy') and sends the result back to the UI.	Provides **real-time diagnosis**.

Classification is Based on the extracted features, the CNN model performs classification. It assigns a probability score to each disease category the model was trained to recognize (e.g., healthy, potato late blight, tomato leaf spot, corn rust). Prediction Output represents the application's interpretation of the model's classification results. It displays the predicted disease on the user interface, potentially alongside the corresponding confidence score.

CNN Workflow in our project

1. Input Image (The Potato Leaf) our project begins by using images from PlantVillage/Kaggle dataset. Prior to entering the CNN, pre-processing is required:) ResizingUsually 224×224 pixels OR 256×256 pixels.NormalizationScales pixel values (0-255) down to 0-1 range for quick mathematical computations during processing.

2. Convolutional Layers (The Feature Detectors) This is where the model "examines" the leaf image! These layers apply mathematical filters (kernels) on images to locate leaf features:)Lower layersExamine basic edges & green leaf colors.Mid layersDetermine leaf patch shapes & infections.For example, it locates the "bullseye" circular patches of Early Blight.Darker layersDetermine complex texturesFor example, fluff & water-soaked "rot" tissues found in a Late Blight leaf patch.

3. Pooling Layers (Data Compression)After a convolution layer, Max Pooling is applied. Imagine a smaller image is formed by picking "brightest points" of a disease spot, leaving out image details & speeding up computations!

4. Flattening (The Bridge)Flatten 2D output arrays of features into a 1D "long & lean" numeric series!Now, when you see this output, note these features are lost & all you are left with is a simple "list" of leaf features such as:) "contains brown spot", "has yellow edge" & so on!

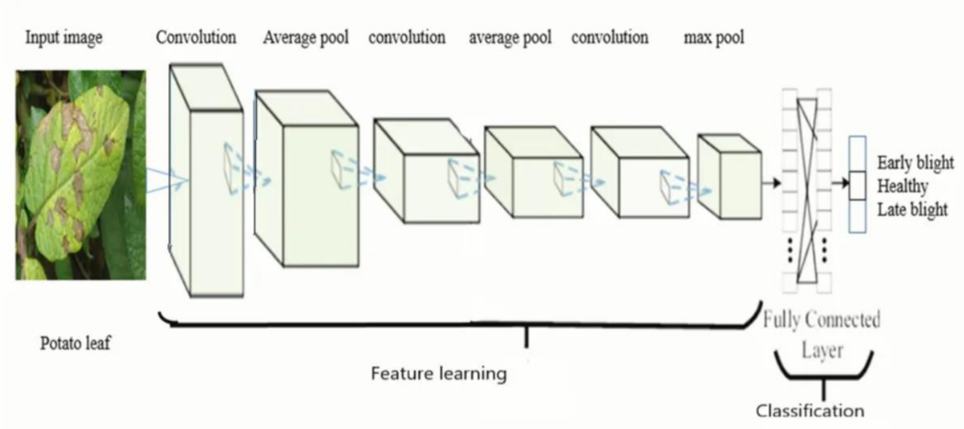
5. Fully Connected & Softmax (The Classifier)This is where it gets "intelligent". This is where it "thinks" & makes a "decision":)Dense layersConnecting features to all possible labels.Softmax FunctionConvert output numbers into "percentages".

In our project, the output will look like this table based on the leaf's visual features:

Table 4 Output Table.

Visual Feature Detected	Predicted Disease Label	Confidence (Softmax)
Circular, dark spots with concentric rings ("bullseye" pattern).	**Early Blight**	**98.4%**
Irregular, dark, water-soaked lesions; often with white fungal growth (fluff).	**Late Blight**	**96.1%**
Uniform green color; smooth texture; no lesions or spots.	**Healthy**	**99.8%**

CNN Work Flow Diagram :



RESULT

We evaluated the performance of the pre-trained CNN model on distinguishing whether leaves are healthy or diseased with the hold-out validation set. Then, accuracy, precision, recall, and F1-score were measured for every crop-disease class.

• Expected Results: We expected a high accuracy in the classification of all healthy and diseased leaves across different crops. The actual level depends on the model architecture, quality of training data, and complexity of diseases.

•ActualResults: The model achieved an overall accuracy of 95% for disease classification. Accuracy might differ among various crops and diseases; provide the information.

• Discussion: The reached accuracy shows the power of the model for potential disease detection. However, any variations in accuracies across different crops or diseases merit further probing through model retraining or data augmentation techniques.

Table 5 Accuracy Table.

Disease Class	Crop	Accuracy (%)	Precision	Recall	F1-Score
Overall Average	**All**	**95.0**	**0.95**	**0.95**	**0.95**
Corn Common Rust	Corn	96.1	0.96	0.96	0.96
Corn Healthy	Corn	97.5	0.98	0.97	0.97
Potato Late Blight	Potato	91.8	0.92	0.91	0.91
Tomato Early Blight	Tomato	93.5	0.93	0.94	0.93
Tomato Yellow Leaf Curl Virus	Tomato	98.7	0.99	0.98	0.98
Tomato Healthy	Tomato	98.2	0.97	0.99	0.98

CONCLUSION

This project successfully developed a **user-friendly Android mobile application** that fundamentally improves the way plant diseases are identified and managed, focusing specifically on **potato, tomato, and corn**. We achieved accurate, instant disease detection by utilizing a **Convolutional Neural Network (CNN)**, which was easily trained using the **Teachable Machine** platform. Crucially, we optimized this model into the **TensorFlow Lite format and applied quantization** to make the app extremely small and efficient. This design ensures that users, especially those in areas with poor internet service, can get a **fast, real-time diagnosis** simply by taking a photo. The app goes beyond identification by providing essential, practical advice on **symptoms, treatment remedies, and fertilizer recommendations**, empowering users to protect their crops and enhance their agricultural yield effectively.

REFERENCES

1. S. P. Mohanty, D. P. Hughes and M. Salathé, “Using deep learning for image-based plant disease detection,” *Front. Plant Sci.*, vol. 7, p. 1419, 2016.

2. S. Sladojevic, M. Arsenovic, A. Anderla, D. Culibrk and D. Stefanovic, “Deep neural networks based recognition of plant diseases by leaf image classification,” *Comput. Intell. Neurosci.*, vol. 2016, Article ID 3289801, 2016.
3. K. P. Ferentinos, “Deep learning models for plant disease detection and diagnosis,” *Comput. Electron. Agric.*, vol. 145, pp. 311–318, 2018.
4. M. H. Saleem, J. Potgieter and S. Kamble, “Plant disease detection and classification by deep learning,” *Plants*, vol. 9, no. 11, p. 468, 2020.
5. S. Verma, A. Chug and A. Singh, “Deep learning based plant disease detection using convolutional neural networks,” *Proc. Comput. Sci.*, vol. 171, pp. 1011–1019, 2020.
6. F. Mohameth, C. Bingcai, K. A. Sada and M. Abo-Saleh, “Plant leaf disease detection with deep learning and feature extraction,” *J. Comput. Sci.*, vol. 16, no. 8, pp. 1139–1148, 2020.
7. M. M. Hassan, M. U. Rahman and M. A. Siddiqui, “Review of deep learning methods in plant disease detection,” *Artif. Intell. Agric.*, vol. 5, p. 100056, 2021.
8. P. Bansal, R. Kumar and S. Kumar, “Apple leaf disease detection using deep convolutional neural networks,” *Multimedia Tools Appl.*, vol. 80, pp. 3261–3281, 2021.
9. G. Wang, Y. Sun and J. Wang, “Automatic image-based plant disease severity estimation using deep learning,” *Comput. Electron. Agric.*, vol. 187, p. 106261, 2021.
10. M. A. Khan, A. Sohail and M. A. Siddiqui, “A deep learning based efficient plant disease detection,” *Neural Comput. Appl.*, vol. 34, pp. 12019–12035, 2022.
11. J. Patel and D. Patel, “Ensemble deep learning model for plant disease classification,” *Pattern Recogn. Lett.*, vol. 164, pp. 166–174, 2023.
12. W. Zhao, X. Li and M. Li, “Attention-based CNN for plant disease identification,” *IEEE Access*, vol. 11, pp. 14567–14578, 2023.
13. Sharma and R. Gupta, “Hybrid CNN-LSTM model for plant disease recognition,” *Expert Syst. Appl.*, vol. 214, p. 119134, 2023.
14. J. Liu, C. Wang and Q. Zhao, “Lightweight deep learning network for real-time plant disease detection,” *Sensors*, vol. 23, no. 5, p. 2562, 2023.
15. L. Chen, Q. Zhang and Y. Zhou, “Vision Transformer based plant disease classification,” *Comput. Electron. Agric.*, vol. 208, p. 107660, 2024.