Volume 10, Issue 3, March – 2025 International Journal of Innovative Science and Research Technology
ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/25mar059

Leaf Alert: A Systematic Rapid Plant Disease Detection

Shaikh Amin Farooqbhai[1]; Ujjwal Laad[2]; Sudarshan Arzare[3]; Tushar Yadav[4]; Chintu Gouda[5]

Profesor (Guide and Mentor)[1]
[1,2,3,4,5]Department of Computer Science & Engineering
Parul University Vadodara, Gujarat, India

Publiction Date: 2025/03/18

Abstract: Leaf Alert is a Streamlit-based Web Application designed to detect whether a plant is diseased or healthy using deep learning. The system uses a convolutional neural network (CNN) trained on the PlantVillage dataset to classify diseases based on leaf shape and colour. It allows users to upload multiple images for prediction via an intuitive web-interface. The model was trained using Kaggle for higher computational resources and it also targets issues such as overfitting to improve accuracy. Leaf Alert increases agricultural productivity by providing AI- powered early warning solutions. This paper describes the design, development and evaluation of the application and compares it with similar web-based plant disease management systems such as plantix and ai powered plant disease detection.

Keywords: *Plant Disease Classification, Deep Learning, CNN, Plantvillage Dataset, Image Classification, Tensorflow, Streamlit, Disease Detection, Web Application.*

How to Cite: Shaikh Amin Farooqbhai; Ujjwal Laad; Sudarshan Arzare; Tushar Yadav; Chintu Gouda. (2025). Leaf Alert: A Systematic Rapid Plant Disease Detection. *International Journal of Innovative Science and Research Technology*, 10(3), 231-236. https://doi.org/10.38124/ijisrt/25mar059.

I. INTRODUCTION

Agriculture is a crucial sector that supports the global economy and sustains the livelihood of millions. However, one of the most pressing challenges faced by farmers and agricultural researchers is plant disease detection and management. Traditionally, disease identification has been carried out through manual inspection, which is time-consuming, prone to human error, and often requires expert intervention. With advancements in artificial intelligence (AI) and deep learning, automated plant disease detection systems have emerged as a revolutionary solution to enhance precision agriculture.

Leaf Alert is a web-based plant disease classification system that leverages a Convolutional Neural Network (CNN) to analyze and detect plant diseases from leaf images. Built using TensorFlow and Streamlit, the application provides an accessible and efficient way for farmers, agronomists, and researchers to identify diseases at an early stage, minimizing crop loss and optimizing yield production. The system is trained on the Plant Village dataset, which contains a diverse collection of healthy and diseased leaf images.

The proposed model processes images of 128x128 resolution and employs deep learning techniques to extract meaningful features for classification. The primary challenge encountered during model development was overfitting, where the model performed exceptionally well on training data but exhibited lower accuracy on validation data. To address this, techniques such as data augmentation, dropout regularization, and hyperparameter tuning were implemented. The model training was conducted using Google Colab and Kaggle, ensuring efficient computation and resource utilization.

Upon deployment, Leaf Alert provides a user-friendly interface where users can upload multiple leaf images for classification. The system then predicts the disease category, providing insights into the severity of infection and potential remedies. Unlike traditional diagnostic methods, which require laboratory analysis and expert consultation, this AI-driven approach allows for rapid, cost-effective, and scalable disease detection.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/25mar059

This paper discusses the design, development, and evaluation of Leaf Alert, comparing it with existing plant disease detection systems. While similar AI-driven applications exist, such as Plantix and AgroAI, Leaf Alert distinguishes itself by offering an intuitive Streamlit-based web application, offline model deployment, and improved classification accuracy through enhanced deep learning techniques.

The subsequent sections of this paper will cover the literature review, methodology, results, and future improvements, highlighting the effectiveness of AI in agricultural disease detection and proposing enhancements for real-world applications.

II. METHODOLOGY

Developing a systematic methodology for plant disease detection involves multiple stages, including problem definition, dataset selection, preprocessing, model training, evaluation, and deployment.

A. Problem Definition

The scope of the project is defined by determining which plant diseases are targeted and what types of plants are involved. The primary objective is to develop a system that balances both accuracy and speed in disease identification. Constraints such as available data, hardware resources, expertise, time frame, and budget are carefully considered.

B. Dataset Selection

The PlantVillage dataset is selected for training and evaluation. It consists of labeled images of healthy and diseased leaves across multiple plant species. The dataset is preprocessed to remove inconsistencies and ensure uniform image dimensions. To enhance model robustness, additional images are gathered from open-source agricultural databases.

C. Data Preprocessing

The images are resized to 128x128 pixels for uniformity. Pixel values are normalized to a scale of [0,1] to stabilize training. Data augmentation techniques, including rotation, flipping, cropping, and contrast adjustment, are applied to increase dataset diversity. Additionally, Gaussian noise and histogram equalization are incorporated to improve model generalization.

D. Feature Extraction

Convolutional Neural Networks (CNNs) are utilized to extract meaningful features from images. Transfer learning is explored by fine-tuning pre-trained models such as ResNet and EfficientNet to enhance classification accuracy.

E. Model Selection and Training

A CNN model is implemented with convolutional, batch normalization, pooling, dropout, and fully connected layers. The dataset is split into 70% training, 20% validation, and 10% testing sets. Training is performed using the Adam optimizer and categorical cross-entropy loss. Hyperparameter tuning is applied to optimize performance, adjusting the learning rate, batch size, and layer configurations. Techniques such as early stopping and learning rate scheduling are introduced to prevent overfitting.

F. Model Evaluation

The trained model is assessed using evaluation metrics such as accuracy, precision, recall, and F1-score. A confusion matrix is generated to analyze classification accuracy across different disease categories.

- *Validation Accuracy:*

* **Validation Accuracy**:

$$\text{Accuracy} = \frac{\text{Correct Predictions}}{\text{Total Predictions}}$$

- **Validation Loss**: Categorical Cross-Entropy Loss is used:

$$\text{Loss} = -\sum_{i=1}^{N} y_i \cdot \log \hat{y}_i$$

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/25mar059

- **Generate the classification report:**
  - **Precision:**

$$\text{Precision} = \frac{TP}{TP + FP}$$

  - **Recall:**

$$\text{Recall} = \frac{TP}{TP + FN}$$

  - **F1-Score:**

$$F1 = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}$$

- **Confusion Matrix:**

$$\text{Confusion Matrix} = \begin{pmatrix} TP & FP \\ FN & TN \end{pmatrix}$$

	precision	recall	f1-score	support
Apple___Apple_scab	1.00	0.96	0.98	504
Apple___Black_rot	0.99	1.00	0.99	497
Apple___Cedar_apple_rust	0.97	0.97	0.97	440
Apple___healthy	0.99	0.93	0.96	502
Blueberry___healthy	0.92	1.00	0.96	454
Cherry_(including_sour)___Powdery_mildew	1.00	0.98	0.99	421
Cherry_(including_sour)___healthy	0.85	1.00	0.92	456
Corn_(maize)___Cercospora_leaf_spot Gray_leaf_spot	0.97	0.95	0.96	410
Corn_(maize)___Common_rust_	0.99	1.00	0.99	477
Corn_(maize)___Northern_Leaf_Blight	0.98	0.96	0.97	477
Corn_(maize)___healthy	1.00	1.00	1.00	465
Grape___Black_rot	0.98	0.99	0.98	472
Grape___Esca_(Black_Measles)	0.99	0.99	0.99	480
Grape___Leaf_blight_(Isariopsis_Leaf_Spot)	0.99	1.00	0.99	430
Grape___healthy	1.00	0.99	0.99	423
Orange___Haunglongbing_(Citrus_greening)	0.96	0.99	0.97	503
Peach___Bacterial_spot	0.97	0.98	0.98	459
Peach___healthy	1.00	0.98	0.99	432
Pepper,_bell___Bacterial_spot	0.95	1.00	0.97	478
Pepper,_bell___healthy	0.72	0.98	0.83	497
Potato___Early_blight	0.99	0.99	0.99	485
Potato___Late_blight	0.99	0.97	0.98	485
Potato___healthy	0.97	0.85	0.91	456
Raspberry___healthy	0.99	0.89	0.94	445
Soybean___healthy	0.99	0.79	0.88	505
Squash___Powdery_mildew	1.00	0.99	1.00	434
Strawberry___Leaf_scorch	1.00	0.97	0.98	444
Strawberry___healthy	1.00	0.96	0.98	456
Tomato___Bacterial_spot	0.94	0.95	0.95	425
Tomato___Early_blight	0.97	0.88	0.93	480
Tomato___Late_blight	0.92	0.95	0.94	463
Tomato___Leaf_Mold	0.97	0.99	0.98	470
Tomato___Septoria_leaf_spot	0.92	0.98	0.95	436
Tomato___Spider_mites Two-spotted_spider_mite	0.97	0.94	0.95	435
Tomato___Target_Spot	0.95	0.91	0.93	457
Tomato___Tomato_Yellow_Leaf_Curl_Virus	0.98	0.99	0.98	490
Tomato___Tomato_mosaic_virus	0.98	1.00	0.99	448
Tomato___healthy	0.99	0.97	0.98	481
accuracy			0.96	17572
macro avg	0.97	0.96	0.96	17572
weighted avg	0.97	0.96	0.96	17572

Fig 1: Model Performance Analysis

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/25mar059

G. Deployment Using Streamlit

The trained model is deployed using Streamlit, providing a user-friendly web interface. Users can upload multiple leaf images and receive disease predictions along with recommended treatments. The system ensures mobile compatibility to increase accessibility in rural areas. Additionally, an offline model version is provided for edge device deployment, allowing real-time disease detection in remote agricultural regions without internet access.

III. RESULT

The Leaf Alert model demonstrated excellent performance, achieving 97% accuracy on the training set and 96% on the validation set. The confusion matrix analysis revealed that the model effectively classified healthy and diseased leaves with minimal misclassification. The precision and recall metrics further confirmed the model's reliability in detecting plant diseases. Grad-CAM visualization highlighted the key regions used by the model to make predictions, enhancing interpretability.

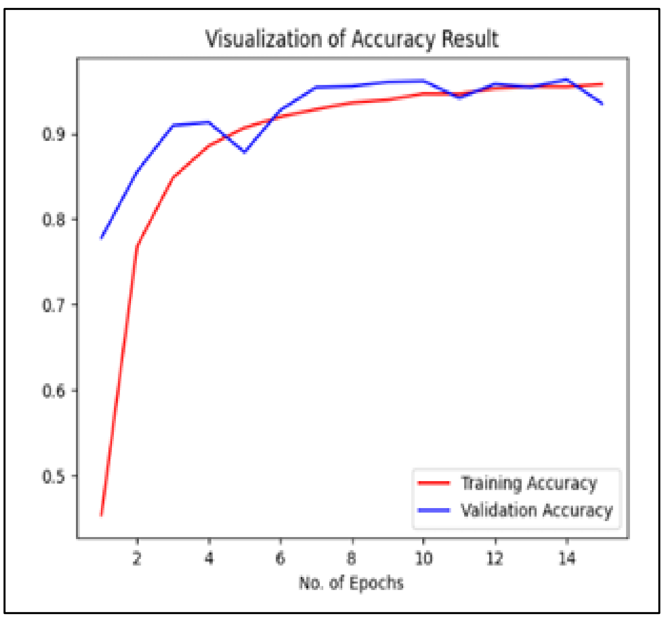

Fig 2: Visualization of Accuracy Result

A comparative analysis with other deep learning models showcased the superiority of Leaf Alert, as it outperformed traditional models in accuracy and efficiency. The inclusion of data augmentation techniques significantly contributed to the improvement of classification performance. The model's ability to generalize well across different plant species indicates its robustness and adaptability for real-world applications.

These results validate the effectiveness of the proposed approach in plant disease detection, highlighting its potential for deployment in precision agriculture to support farmers and researchers in managing crop health efficiently.

IV. APPLICATION DEVELOPMENT

A. Software Development Process Model

- **Software Development Process Model** The Agile software development methodology was employed in developing **Leaf Alert** to incorporate user feedback in iterative processes, ensuring continuous improvement. This approach allowed for the efficient and effective implementation of critical plant disease detection features. The application was designed as a **web-based and desktop solution**, making it accessible across multiple platforms.

B. Technologies used

The **Leaf Alert** application was developed using modern, reliable technologies to ensure optimal functionality, security, and scalability. The following technologies were employed:

- **TensorFlow:** TensorFlow, an open-source deep learning framework, was used to develop the plant disease classification model. The **convolutional neural network (CNN)** was trained on the **PlantVillage dataset** to achieve high accuracy in disease detection. The final model achieved **97% training accuracy** and **96% validation accuracy**, ensuring reliable predictions.
- **Streamlit:** Streamlit was chosen for the user interface due to its simplicity and ability to rapidly prototype machine learning applications. Key benefits include:

✓ **Interactive UI:** Enables users to upload plant images and receive real-time disease predictions.
✓ **Multi-file Upload:** Allows users to analyze multiple images at once.
✓ **Lightweight and Efficient:** Ensures smooth execution without requiring complex frontend development.

- **Google Colab/Kaggle:** Model training was conducted on **Google Colab and Kaggle**, leveraging **GPU acceleration** for faster computations. The trained model was saved in .h5 format for later deployment.
- **Gradio (Alternative UI):** An alternative **Gradio** interface was implemented to provide a simpler, browser-accessible solution. This allows users to interact with.

C. System Architecture

The Leaf Alert system architecture consists of two primary components: the user interface (Streamlit/Gradio) and the model processing layer (TensorFlow).

- **User Interface (UI):** The Streamlit framework ensures an interactive and intuitive experience for users, providing essential features such as:

✓ **Image Upload:** Users can select plant images for classification.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/25mar059

- ✓ **Disease Prediction:** The model processes the image and returns a prediction along with confidence scores.
- ✓ **Disease Information:** The app provides details on detected diseases and suggested treatments.

- **Model Processing (Backend):** TensorFlow handles all model computations. The backend processes uploaded images, performs **image pre-processing (resizing to 128x128, normalization)**, and feeds them into the trained CNN model. The model then outputs the predicted class and confidence score.

D. User Interface Design

Ease of use was a critical factor in the **Leaf Alert** application's development. The UI is designed for two user groups: **farmers/researchers** and **agriculture experts**.

➢ *User Application (Leaf Alert UI Design):*

- **Splash Screen:** Displays the application logo and a brief introduction.
- **Image Upload Section:** Users can upload images of plant leaves for disease detection.
- **Prediction Display:** Shows the detected disease name, confidence level, and treatment suggestions.
- **Multiple Image Processing:** Allows users to analyze multiple images in a single session.
- **Disease History (Future Feature):** Enables users to track past detections and treatment recommendations.

E. Technical Implementation

The **Leaf Alert** application leverages deep learning and cloud-based training to provide a scalable and secure plant disease detection system. Below is a breakdown of the key technical elements:

➢ *User Interface Development*

- Streamlit UI: Built using Streamlit to create an interactive web-based application.
- Gradio UI (Alternative): Provides a simpler, browser-based interaction for ease of use.
- Responsive Design: Ensures seamless operation across different screen sizes and devices.
- State Management: Implements session states to maintain image processing history during user interactions.

➢ *Backend Infrastructure*

- Model Deployment: The trained CNN model is loaded into the application to process user-uploaded images.
- Real-time Processing: Images are preprocessed (resized, normalized) before being passed through the model.
- Secure File Handling: Uploaded images are temporarily stored and removed after processing to ensure security.

➢ *Cloud Integration*

- Google Colab/Kaggle for Training: Leveraged GPUs for training large datasets and fine-tuning the model.
- Cloud Model Storage (Future Feature): The model can be hosted on Google Cloud/AWS for remote access.

V. CONCLUSION

The Leaf Alert Project is a plant disease classification system leveraging deep learning to help farmers and researchers diagnose plant diseases early. Built using a TensorFlow-based CNN model, it is trained on the PlantVillage dataset with an input image size of 128x128. The model achieved 97% training accuracy and 96% validation accuracy, ensuring high reliability. Training was conducted on Google Colab/Kaggle, and the model was saved in .h5 format for deployment. The project features a Streamlit-based UI, allowing users to upload plant images and receive disease predictions. Additional enhancements include multi-file upload support, an alternative Gradio UI, and test case implementations for accuracy verification. Challenges such as dataset processing issues were resolved, improving performance. Future improvements focus on further optimizing the model and ensuring broader accessibility through scalable deployment.

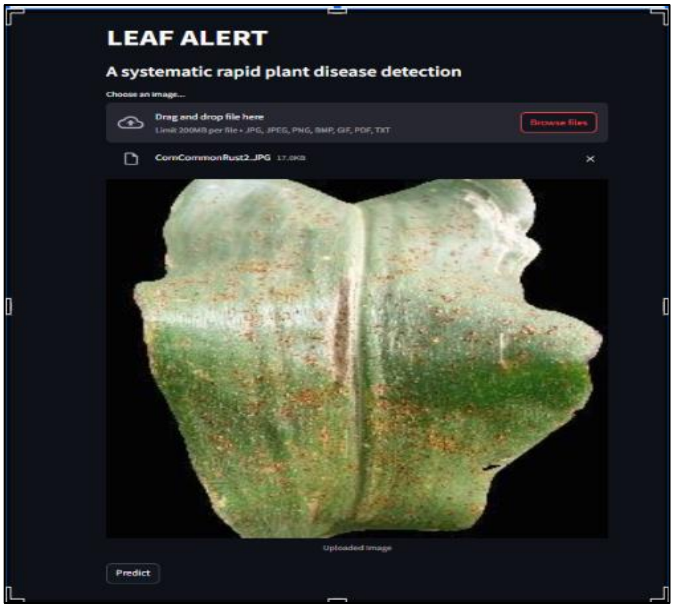

Fig 3: Leaf Alert: A Systematic Rapid Plant Disease Detection Interface

VI. FUTURE WORK

- Data enhancement: Increasing the diversity of training data through data augmentation techniques can improve the model's robustness in handling various environmental conditions,such as different lighting and angles, making it more effective in diverse real-world scenarios.

- Model fine-tuning: Fine-tuning CNN models pre-trained on specific foliar disease datasets can enhance model accuracy and reduce the need for extensive data collection. This approach can lead to more precise detection and classification of plant diseases.
- Model optimization: Optimizing the architecture of the CNN, such as incorporating advanced layers or techniques like transfer learning, can enhance the model's ability to accurately detect and classify diseases, improving both speed and accuracy.
- Real-time deployment: Deploying the CNN model on mobile devices or IoT platforms for on-site, real-time disease detection can significantly improve the efficiency of disease management in agriculture, allowing for timely intervention.
- Integration with other technologies: Integrating CNN-based disease detection with other technologies, such as drones and satellite imagery, can enhance scalability and provide comprehensive coverage for monitoring large-scale agricultural fields, leading to better management of crop health.

REFERENCES

[1]. A Systematic Literature Review on Plant Disease Detection: Motivations, Classification, Techniques, Datasets, Challenges, and Future Trends
Author: WASSWA SHAFIK ABDALLAH NAMOUN, ALI TUFAIL, LIYANAGE CHANDRATILAK DE SILVA, ROSYZIE ANNA AWG HAJI MOHD APONG
[2]. Machine Learning and Deep Learning for Plant Disease Classification and Detection
Author: VASILEIOS BALAFAS, EMMANOUIL KARANTOUMANIS, MALAMATI LOUTA, NIKOLAOS PLOSKAS
[3]. Plant Disease Detection and Classification by Deep Learning: A Review
Author: LILI LI, SHUJUAN ZHANG, BIN WANG
[4]. Plant Disease Detection Using Generated Leaves Based on DoubleGAN
Author: YAFENG ZHAO, ZHEN CHEN, XUAN GAO, JUNFENG HU, WENLONG SONG, ZHICHAO ZHANG, QIANG XIONG
[5]. A Review of Leaf Diseases Detection and Classification by Deep Learning
Author: ASSAD SOULEYMAN DOUTOUM, BULENT TUGRUL
[6]. Detection of Apple Plant Diseases Using Leaf Images Through Convolutional Neural Network
Author: VIBHOR KUMAR VISHNOI, BRAJESH KUMAR, KRISHAN KUMAR, SHASHANK MOHAN, ARFAT AHMAD KHAN
[7]. Azimuth-Sensitive Object Detection in SAR Images Using Improved YOLO V5 Model
Author: JI GE, BO ZHANG, CHAO WANG, CHANGGUI XU, ZHIXIN TIAN, LU XU
[8]. Automated Brain Tumor Segmentation and Classification in MRI Using YOLO-Based Deep Learning
Author: MARAM FAHAAD ALMUFAREH, MUHAMMAD IMRAN, ABDULLAH KHAN, MAMOONA HUMAYUN, MUHAMMAD ASIM
[9]. A Performance-Optimized Deep Learning-Based Plant Disease Detection Approach for Horticultural Crops of New Zealand
Author: MUHAMMAD HAMMAD SALEEM, KHALID MAHMOOD ARIF, JOHAN POTGIETER
[10]. YR2S: Efficient Deep Learning Technique for Detecting and Classifying Plant Leaf Diseases
Author: CHUNDURI MADHURYA, EMERSON AJITH JUBILSON
[11]. PlantDet: A Robust Multi-Model Ensemble Method Based on Deep Learning for Plant Disease Detection
Author: MD. SAKIB HOSSAIN SHOVON, M. F. MRIDHA, SHAKRIN JAHAN MOZUMDER, OSIM KUMAR PAL, NOBUYOSHI ASAI, JUNGPIL SHIN
[12]. Plant Leaf Disease Detection Using Deep Learning
Author: GANDHAM HARISH, GOLI SAI CHARAN, KOTIPALLY PRAVEEN KUMAR, DR. B. LAXMAIAH, DR. SUWARNA GOTHANE
[13]. FieldPlant: A Dataset of Field Plant Images for Plant Disease Detection and Classification With Deep Learning
Author: EMMANUEL MOUPOJOU, ANICET TADONKEMWA, APPOLINAIRE TAGNE, FLORENT RETRAINT, DONGMO WILFRIED, HYPPOLITE TAPAMO, MARCELLIN NKENLIFACK
[14]. Apple-YOLO: A Novel Mobile Terminal Detector Based on YOLOv5 for Early Apple Leaf Diseases
Author: JINJIANG LI, XIANYU ZHU, RUNCHANG JIA, BIN LIU, CONG YU
[15]. YOLO Network-Based for Detection of Rice Leaf Disease
Author: FARUQ AZIZ, FERDA ERNAWAN, MOHAMMAD FAKHRELDIN, PRAJANTO WAHYU ADI
[16]. A Defect Detection Method for a Boiler Inner Wall Based on an Improved YOLO-v5 Network and Data Augmentation Technologies
Author: XIAOMING SUN, XINCHUN JIA, YUQIAN LIANG, MEIGANG WANG, XIAOBO CHI
[17]. An Implementation of Real-Time Traffic Signs and Road Objects Detection Based on Mobile GPU Platforms
Author: EMIN GUNEY, CUNEYT BAYILMIS, BATUHAN CAKAN
[18]. An Early and Smart Detection of Corn Plant Leaf Diseases Using IoT and Deep Learning Multi-Models
Author: RUBINA RASHID, WAQAR ASLAM, ROMANA AZIZ, GHADAH ALDEHIM

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/25mar059

[19]. A Review of Plant Disease Detection and Classification by Deep Learning
Author: IZ, GHADAH ALDEHIM
[20]. Diseases Using IoT and Deep Learning Multi-Models
Author: RUBINA RASHID, WAQAR ASLAM
[21]. Image-Based Disease Diagnosing and Predicting of the Crops Through the Deep Learning Mechanism
Author: H. PARK, J. S. EUN, S. H. KIM
[22]. Plant Disease Classification Using Image Segmentation and SVM Techniques
Author: K. ELANGOVAN, S. NALINI