International Journal of Environmental Sciences
ISSN: 2229-7359
Vol. 11 No. 9s, 2025
https://www.theaspd.com/ijes.php

Interpretable Deep Learning for Sustainable Agriculture: CNN and LIME-Based Plant Disease Diagnosis

Mohit Arora,
Department of Computer Science & Engineering, Lovely Professional University, Punjab, India
mohit.15980@lpu.co.in

Abhishek Santra,
Department of Computer Science & Engineering, Lovely Professional University, Punjab, India
abhishek128642@gmail.com

Dharmendra Pathak,
Department of Computer Science & Engineering, Lovely Professional University, Punjab, India,
dharmendra.32553@lpu.co.in

Dr. Manoj Agrawal,
Associate Professor, Department of Computer Science & Engineering, Chameli Devi Group of Institutions Indore, Madhya Pradesh, India
drmanojagrawal79@gmail.com

Shivali Chopra,
Department of Computer Science & Engineering, Lovely Professional University, Punjab, India
shivali.19259@lpu.co.in

Dr. Harshita Vachhani,
Pratibha College of Commerce & Computer Studies, Chinchwad, Pune, Maharashtra – 411019, India,
profharshita@gmail.com

Abstract— The manual inspection forms the bulk of conventional plant disease identification methods, but it is inefficient, unreliable, and time-consuming for large-scale agricultural farms. Though they are promising, numerous automated systems that are currently employed struggle to be applied in practice due to their transparent decision-making processes and low capacity to adapt to real-world field conditions. This research places emphasis on readability in predictions while developing a clear and reliable deep learning model to diagnose diseases among 38 plant species. The objective of the research was to counteract the increasing threat of food insecurity, which is being exacerbated by population growth and climate change. Methods like picture augmentation and regularization ensure the model's adaptability to different environmental conditions. Explanation tools visually highlight specific aspects of disease-related images, which is crucial to adoption and boosts user trust. Agronomic information is also aligned with visual explanations. Through the reduction of unnecessary use of pesticides, the strategy not only supports sustainable agricultural practices but also propels global efforts in equitable agricultural innovation, poverty reduction, and climate resilience. This study presents a CNN-based model for multi-class plant disease classification across 38 categories using an augmented dataset of leaf images. The model achieved a high-test accuracy of 97.96% and demonstrated strong generalization with minimal overfitting. LIME-based explainability validated the model's agronomic relevance, enhancing its trust and applicability in real-world agricultural diagnostics.

Keywords— Convolutional Neural Networks (CNN), Deep learning, Image Processing, Local Interpretable Model-agnostic Explanations (LIME), Plant Disease Detection.

INTRODUCTION

A global threat to the world which includes the necessary amount of agriculture that requires protection from plant diseases. Failing to do so results in economic problems, food production capacity and lower crop yields. Diagnosing timely with accurate measures is highly essential to stop the spreading of such diseases caused by various microbial beings. Techniques that are conventional such as DNA testing in lab or visually inspecting the crops by agronomists who visually examine the plants are employed. Often these techniques end up with being highly time-consuming, labor-intensive and difficult for small-scale farmers too in some isolated locations when used [1]. Artificial intelligence (AI) has become a revolutionary tool for automating disease detection in recent years. In particular, visual patterns linked to infected plant tissues have been successfully analyzed by CNNs as shown in Fig. 1. However, there are three main obstacles to its practical implementation: (1) an architecture that can achieve high accuracy while being computationally efficient; (2) overfitting and associated issues like imbalanced or limited datasets; and (3) the opaque nature of deep learning, which is essentially a "black box" for many algorithms, which undermines practitioner buy-in and trust, particularly in agricultural systems. The research was done to highlight and resolve the problems faced as discussed above by building and employing case-specific interpretable deep-learning framework for plant disease classification, wherein a particular CNN architecture is designed for high productivity and interpretability from an agricultural perspective. The New Plant Diseases Dataset contains high-resolution images that comprises 38 disease categories across 14 plant species [2].

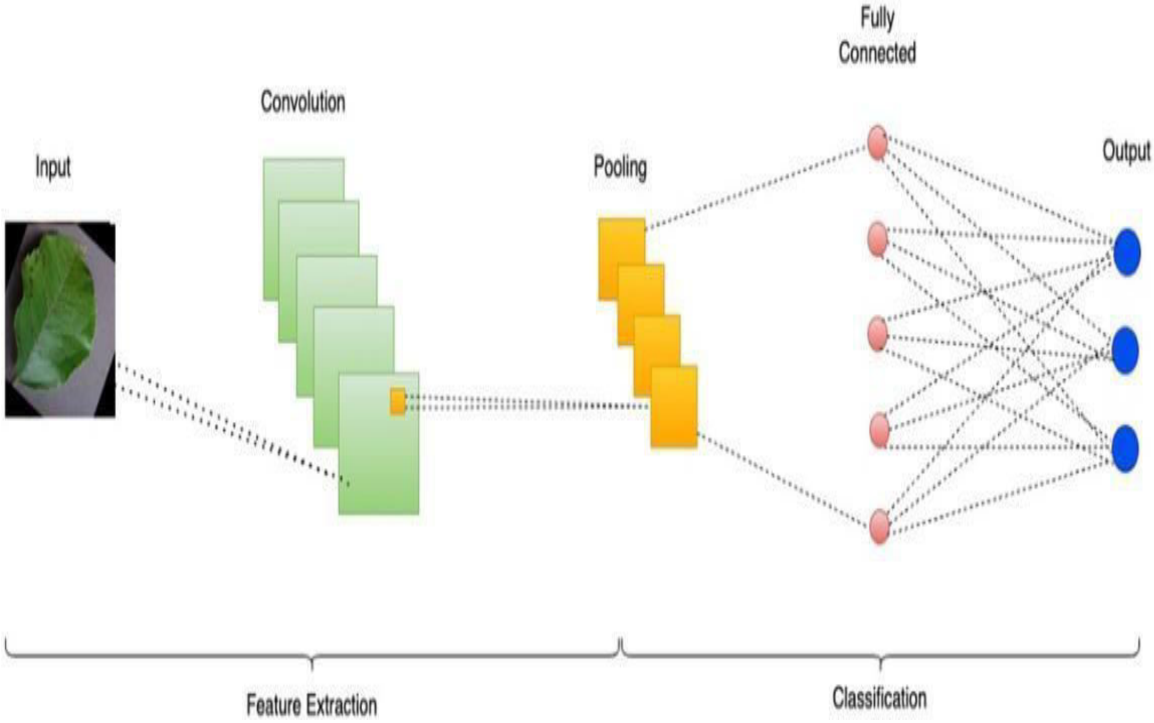


Fig. 1: CNN Model Architecture for Plant Disease Classification

The model's design incorporates hierarchical feature extraction, beginning with shallow convolutional layers to capture low-level textures (e.g., leaf veins, discoloration) and progressively deepening to abstract representations (e.g., lesion morphology, spatial distribution of symptoms). To counteract overfitting, which commonly plagues models trained on biologically variable and imbalanced datasets, we integrate batch normalization to stabilize layer activations, L2 regularization to penalize overly complex weight configurations, and targeted dropout layers (25–50%) to prevent co-adaptation of neurons. Complementing these architectural innovations, we employ aggressive data augmentation to simulate the diversity of field conditions, such as uneven lighting, occlusions from debris, and variations in leaf orientation [3]. This approach ensures robustness to the inherent variability encountered in real-world agricultural environments. A key aspect of our work is the integration of explainable AI techniques to illuminate the model's decision-making process. Despite their superiority in classification tasks, traditional CNNs frequently produce predictions that are difficult to interpret and also difficult for users to trust results that deviate from their expert knowledge. With LIME, a post-hoc interpretability framework that produces readable saliency maps by pixel-level explanation, we close the gap between prediction and post-hoc explanations. The saliency maps show the areas of input photos that the model deemed to be the most diagnostically useful, such as fungal spots, chlorotic borders, or necrotic patches [4]. A natural biological basis for justification is provided by LIME, which enables us to validate our model in relation to recognized clinical symptoms, such as angular lesions suggestive of bacterial leaf spot or circular ring patterns indicating late blight. The model's generalize performance across common and uncommon disease classes is demonstrated by its 95.44% test accuracy, 95.59% macro-averaged precision, and 95.41% recall scores. A finer resolution of the misclassifications made possible by a confusion matrix shows that errors primarily occur between phenotypically similar conditions (e.g., early vs. late stage of the same disease) rather than among completely different pathologies. The model also learns to detect semantically meaningful features. Routines on LIME explanations reliably capture places such as the necrotic water-soaked lesion patterns in Alternaria solani or the powdery patches associated with Erysiphe cichoracearum. These results demonstrate that the model can mimic (or even outperform) the diagnostic heuristics of human professionals at large scale and speed. Beyond technical breakthroughs, this research shows how human-centric AI is transforming sustainable agriculture. By combining interpretability and high accuracy, our approach eliminates a significant obstacle to acceptance: stakeholders' reluctance to trust opaque systems with important choices. Farmers might make quick diagnoses in the field if they had smartphone-based versions of this technology. This could lead to prompt actions like crop rotation or focused pesticide application [5]. By combining the model's predictions with geographic data, agricultural extension agencies may monitor disease outbreaks at the regional level and allocate resources accordingly. Moreover, the principles of being efficient and modular in designing in TensorFlow inference (optimized for interoperability with edge devices) helps in enabling a wide range of applicability in such a resource-constrained environments and also it helps in promoting the sociotechnical movement of democratizing AI for social welfare related programs. The usage of high multimodal AI systems to enhance precision agriculture is made into a reality by completing this research. Iterations in future might employ federated learning to collect data from several locations with respect to absolute maintenance of data privacy and also the hyperspectral imaging for detection of diseases before the possible symptoms appear. We as humans are one step closer to a future in which artificial intelligence (AI) collaborates with humans to safeguard food security, assist farmers, and advance ecological sustainability by improving the precision and comprehension of plant disease diagnostics.

LITERATURE REVIEW

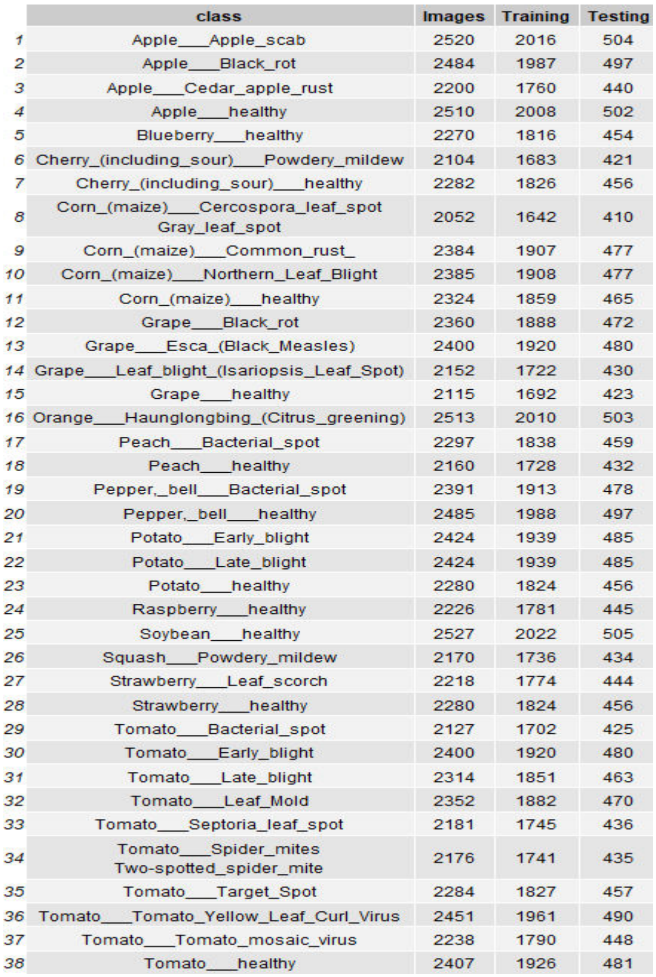
Amrita S. Tulshan (2019) uses a technique which combines image processing, k-means segmentation, with KNN and GLCM feature extraction. An accuracy of 97.6% was achieved and the system outperformed SVM in identification of illnesses including impact areas by analyzing grayscale- converted RGB images in a dataset containing 75 leaves. In order to achieve an accuracy of 98.56% [6]. Meghana Govardhan (2019) proposes a Random Forest based algorithm that achieved an accuracy of 95.2% for tomato plant diseases that includes Mosaic Virus and Early Blight. Dataset used is being divided into 90% training and 10% testing using k-fold cross validation. It contrasts ML methods and emphasizes Random Forest's higher accuracy in disease prediction. It is implemented in Python [7]. Md. Jahid Hasan (2020) uses CNN for feature extraction and BiLSTM for contextual correlation analysis in his CNN-BiLSTM hybrid model, which achieves accuracy of 99.02% in classification and detection of nine maize diseases. Computer vision is used to address pathogen-related threats to maize production and deep learning for automation. By capturing intricate leaf patterns and spatial-temporal linkages, the model makes it possible to distinguish between healthy and ill plants through experimental testing [8].S. Santhana Hari (2019) proposes PDDNN, a customized CNN model for classifying plant diseases, attaining 86% accuracy across 10 categories (maize, grape, apple, tomato) and 97. 5% for healthy apple plants. Convolutional layers, max pooling, and dropout/Batch Normalization was utilized on plant village and customized datasets to mitigate overfitting and reduce computation time. confusion matrix is used to tackle issues such as varying lighting/backgrounds. PDDNN performs better than MobileNet in real-world scenarios for early disease detection [9]. D. Jayakumar (2020) uses a stacked RNN technique for classifying melon leaf illnesses using image segmentation and deep learning that combines K-means clustering and a Stacked RNN. TensorFlow/Keras-built Stacked RNN classifies diseases through steps including segmentation, feature extraction, and image pre-processing, whilst K-means separates diseased regions. The model enables quick detection by achieving high accuracy with little training/validation loss and little processing time [10]. Sristy Saha (2021) proposes Random Forest algorithm achieving an accuracy of 91.47% to find major rice diseases. 528 resized (300x300) grayscale images are being utilized as the approach applies intensity moments for feature extraction following image preprocessing. Early disease detection in rice is achieved by acquisition, pre-processing , classification and execution in Matlab [11].Akash Sirohi (2021) proposes a hybrid deep learning model for early detection that uses stacking ensemble learning with a combination of VGG-16 and MobileNet for classification of sunflower illnesses like Phoma blight, Downy mildew, Verticillium wilt, Alternaria leaf blight with an accuracy of 89.2%. The model performs better than AlexNet and DenseNet121 [12].Saurav Sagar(2024) survey highlights Explainable AI techniques (SHAP, LIME, Grad-CAM, Interpret ML) to improve transparency in deep learning models for plant leaf disease detection. It evaluates traditional and deep learning methods as deep learning boosts efficiency, its opaque decision-making poses challenges, reducing user trust. XAI bridges this gap by making AI outputs interpretable, aiding stakeholders in understanding predictions. The study underscores the urgency of accurate, unbiased interpretations to ensure reliable disease diagnosis [13]. Minah Jung (2023) developed a CNN-based automated system for early plant disease detection, achieving 97.09% accuracy in classifying crops and diseases using pre-trained models. EfficientNet excelled, with 99.33% accuracy in crop identification (bell pepper, potato, tomato) and 97.09% in tomato disease classification. The three-step model detects crops, disease presence, and type, while labeling non-model species as ‘unknown’—enhanced by added data. Validation showed higher accuracy with full-leaf images over lesion-cropped ones. Designed for scalability, the system aims to support smart farming through rapid disease management and industrial integration [14].Jayraj Chopda (2023) introduces a Decision Tree Classifier-based system to predict cotton crop diseases using environmental parameters like temperature and soil moisture. The Android app-driven approach integrates multi-source data for automated, real-time disease prediction, overcoming traditional methods' inefficiencies. Targeting Anthracnose, Areolate (Greymildew), and Wilt, it analyzes user inputs against datasets to deliver predictions. Pending app development, the system aims to enhance agricultural decision-making through accessible smart farming solutions [15]. Anwar Rifa’i (2020) developed a fuzzy logic system for rapid rice disease detection using 120 plant images, achieving 94.8% training and 91.7% testing accuracy. The method integrates Gauss membership fuzzification, Sugeno inference, and weighted average defuzzification with 96 rules to analyze 10 image-derived inputs. Addressing agricultural yield losses, it offers a cost-effective alternative to lab-based diagnosis by combining image preprocessing, feature extraction, and a user-friendly GUI for farmers. The system enhances decision-making for timely disease identification and treatment [16].Monzurul Islam (2017) proposes an SVM-based method for automated potato disease detection using the 'Plant Village' image database, achieving 95% accuracy on 300 leaf images. The system employs image segmentation to isolate diseased regions, leveraging color and texture features (via GLCM) for classification. It uses a multiclass SVM with a linear kernel, attaining 95% testing accuracy and 93.7% 5-fold cross-validation accuracy [17]. B. Rajesh (2020) proposes a decision tree-based method for leaf disease identification, achieving over 95% accuracy with automated image processing (pre- processing, segmentation, feature extraction) to replace manual visual inspections. Targeting agriculture- dependent regions, the system enables early disease detection using machine learning trained on 1,000+ leaf images, reducing time and productivity losses. It analyzes morphological leaf traits to classify diseases and suggest remedies, outperforming existing systems in speed and precision [18]. Pranjali B. Padol (2016) proposes an SVM-based image processing method to detect grape leaf diseases like Downy Mildew (93.33% accuracy) and Powdery Mildew (83.33% accuracy), achieving 88.89% overall accuracy. The approach uses K-means clustering for disease segmentation and extracts color/texture features for classification. Targeting diseases causing 10-30% production losses in India, the method involves image acquisition, preprocessing (resizing, thresholding, noise reduction), and feature extraction. Linear SVM classifies diseases by analyzing shape, color, and texture properties [19]. Shyamtanu Bhowmik (2020) developed a CNN-based model to detect tea leaf diseases (Black Rot, Rust) impacting India’s food security, achieving 95.93% precision with low computational complexity. The system uses 2,341 images (healthy/diseased) across three categories, processed through training and testing phases. Optimized with Adam (0.0001 learning rate), the model trained for 22 epochs (batch size 15), reaching 90% accuracy when dropout increased to 0.25. Focused on rapid identification of unhealthy leaves, it addresses threats to tea crop productivity, a vital Indian commodity [20]. Muhammad Thaqif bin Mohamad Azmi (2013) suggests a method for detecting orchid diseases that analyzes leaf characteristics including centroid, area, and spots using fuzzy logic and image processing techniques. The technique uses MATLAB to process 80 photos of sick leaves, using fuzzification, inference, and defuzzification to convert the retrieved data into health status. Using fuzzy logic, leaves are categorized as either healthy or unhealthy based on the presence of illness. Despite being highly efficient, the noise while capturing images affects the accuracy and must be improved. The study seems pretty viable, still emphasizes the necessary requirements of better noise control and optimizing the system [21]. Eftekhar Hossain (2019) proposes a K-nearest neighbor (KNN) classifier technique that is well known for its high effectiveness and being simple. For segmenting colors and analyzing textures by using Gray Level Co-occurrence Matrix (GLCM) information the procedure achieves an outstanding accuracy of 96.76% after being tested on a collection of data impacted by illnesses which includes anthracnose and Alternaria alternata. This approach is well efficient and validates the performance indicators [22]. The reviewed papers show various machine learning and deep learning approaches for efficient plant disease detection across various crops. From conventional techniques to more advanced architectures such as CNN, BiLSTM, and hybrid deep learning ensembles (e.g., VGG-16 + MobileNet) are also discussed, achieving impressive accuracies ranging from ~89% to over 99%. Recent trends highlight the importance of Explainable AI (XAI) for model interpretability and integration with real-time mobile applications in the agricultural domain.

PROPOSED METHODOLOGY

This section particularly represents the in-depth process that was followed for preparing the dataset suitable for the model architecture. Model performance metrics that was achieved by our model. Also, it contains the explanation and visualization of the output.

A. Dataset Description

The New Plant Diseases Dataset (Augmented) is a significant advance in the field of agricultural machine learning research, offering a robust, high-quality resource for the automated identification and classification of plant diseases. It contains thousands of labeled images from different plant species such as major crops like rice, wheat, maize, and tomato—and uses advanced augmentation techniques such as rotation, flipping, color changes, and synthetic noise addition to enhance variability and simulate real-world environmental conditions. The original dataset is available on this GitHub repository. The dataset comprises around 86K RGB images of diseased and healthy crop leaves, labeled into 38 categories as shown in Fig. 2. The train-test ratio is set to 80/20 for the used dataset. Furthermore 33 new test images are generated for prediction purposes. Each image is thoroughly annotated by agronomy experts, ensuring precise disease identification, and the inclusion of healthy plant samples facilitates binary classification tasks. The dataset comprises high-resolution images whose sizes vary from (224x224 to 1024x1024 pixels), facilitating both deep and light-weight CNNs, and facilitating research on the compromise between accuracy and model efficiency from (224x224 to 1024x1024 pixels), supporting both deep and lightweight convolutional neural networks (CNNs), and supporting studies exploring the trade-off between accuracy and model robustness, generalizing the ability of detection of diseases. Data imbalance which poses a great threat to model performance is solved by using this dataset.



	class	Images	Training	Testing
1	Apple___Apple_scab	2520	2016	504
2	Apple___Black_rot	2484	1987	497
3	Apple___Cedar_apple_rust	2200	1760	440
4	Apple___healthy	2510	2008	502
5	Blueberry___healthy	2270	1816	454
6	Cherry_(including_sour)___Powdery_mildew	2104	1683	421
7	Cherry_(including_sour)___healthy	2282	1826	456
8	Corn_(maize)___Cercospora_leaf_spot Gray_leaf_spot	2052	1642	410
9	Corn_(maize)___Common_rust_	2384	1907	477
10	Corn_(maize)___Northern_Leaf_Blight	2385	1908	477
11	Corn_(maize)___healthy	2324	1859	465
12	Grape___Black_rot	2360	1888	472
13	Grape___Esca_(Black_Measles)	2400	1920	480
14	Grape___Leaf_blight_(Isariopsis_Leaf_Spot)	2152	1722	430
15	Grape___healthy	2115	1692	423
16	Orange___Haunglongbing_(Citrus_greening)	2513	2010	503
17	Peach___Bacterial_spot	2297	1838	459
18	Peach___healthy	2160	1728	432
19	Pepper,_bell___Bacterial_spot	2391	1913	478
20	Pepper,_bell___healthy	2485	1988	497
21	Potato___Early_blight	2424	1939	485
22	Potato___Late_blight	2424	1939	485
23	Potato___healthy	2280	1824	456
24	Raspberry___healthy	2226	1781	445
25	Soybean___healthy	2527	2022	505
26	Squash___Powdery_mildew	2170	1736	434
27	Strawberry___Leaf_scorch	2218	1774	444
28	Strawberry___healthy	2280	1824	456
29	Tomato___Bacterial_spot	2127	1702	425
30	Tomato___Early_blight	2400	1920	480
31	Tomato___Late_blight	2314	1851	463
32	Tomato___Leaf_Mold	2352	1882	470
33	Tomato___Septoria_leaf_spot	2181	1745	436
34	Tomato___Spider_mites Two-spotted_spider_mite	2176	1741	435
35	Tomato___Target_Spot	2284	1827	457
36	Tomato___Tomato_Yellow_Leaf_Curl_Virus	2451	1961	490
37	Tomato___Tomato_mosaic_virus	2238	1790	448
38	Tomato___healthy	2407	1926	481

Fig. 2: Categories of Crop Leaves

B. Data Preprocessing

Initially, a standardized framework was used for preprocessing of the images of the plant disease dataset. It has also been described elsewhere, to further facilitate easy management. Thus, ensuring a uniform motion and to make sure its effectivity in feature extraction, all images had to be resized to a particular common format of 256×256 pixels. Fig. 3 shows the sample images from the dataset.

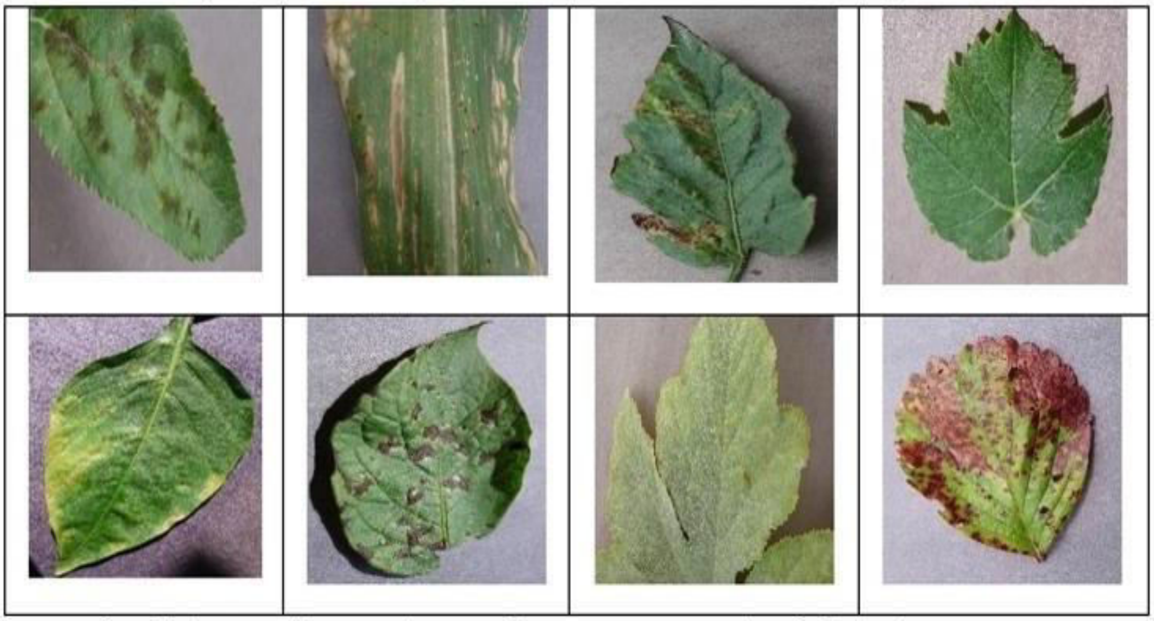


Fig. 3: Sample Images from the Dataset

The training and testing subsets had to be arranged and organized into directories, whereas numerical labels were assigned by the disease categories to simplify processing that Analyzed various batches that contained 32 images thus ensuring that the computation was highly maintained at an efficient pace while maximizing the system performance. Pixel values had to be automatically adjusted by themselves to fall within the [0, 1] range to ensure the standard input without manually intervening.

Randomization for reproducibility was set in this research at specific parameters during dataset partitioning. Datasets used for processing had variation in lighting and orientation to mimic real-world conditions but hence also streamlined the pipeline by preventing extra transformations. The training and testing iterators can be easily integrated into image analysis models with optimal resource allocation. The approach focuses on reducing the complexity of preprocessing while retaining essential spatial information required for disease pattern detection. The approach is designed to be flexible and scalable to larger datasets and multispectral imaging systems.

C. Data Augmentation

Image variability was systematically increased to make the dataset more robust for computational analysis. Pixel intensities were rescaled to the range [0, 1] to ensure uniform input normalization. Spatial and tonal alterations were made to the training data, which now includes random horizontal/vertical flips for different orientations of plants, rotations (up to 20%) allowed as a simulation of angled captures, and zoom variations (±20%) that represent different distances from where the photo is taken or leaves being only partially visible) as possible photo-takiing distance or partial leaf visibility scenarios. Zoom variations imitated camera distance or partial leaf visibility scenarios. Contrast variations were added here as well, meant to emulate variable field lighting conditions. These modifications were applied on-the-fly during training in order not to permanently alter the original dataset. The test data underwent only normalization. To maximize computational efficiency, transformations were executed in batches in parallel with data loading. This approach augmented the flexibility of the dataset by simulating real-world imaging challenges like plant position or environmental lighting changes. It was necessary to carefully limit parameters for adjusting variability while still incorporating significant diagnostic features. The framework focused on reproducing natural agricultural conditions enabling reliable analysis of plant patterns related to specific diseases.

D. Model Flowchart

The below Fig. 4 represents a comprehensive pipeline for plant disease detection using CNNs enhanced with LIME for explainable AI. The system preprocesses a 256×256×3 input image through several convolutional layers, extracting low- to high-level features, including edges, patterns, and disease-specific traits, while adding max pooling, batch normalization, and dropout to prevent overfitting. The final classification is output through fully connected layers with softmax activation over disease classes. For interpretability, the image is divided into superpixels and 2000 customized versions are fed to the model. LIME then fits a local interpretable model to determine which superpixels most suitable for the CNN's prediction. The process ends by identifying important segments that visually explain the model's decision, thereby making the AI's output transparent and actionable for end users.

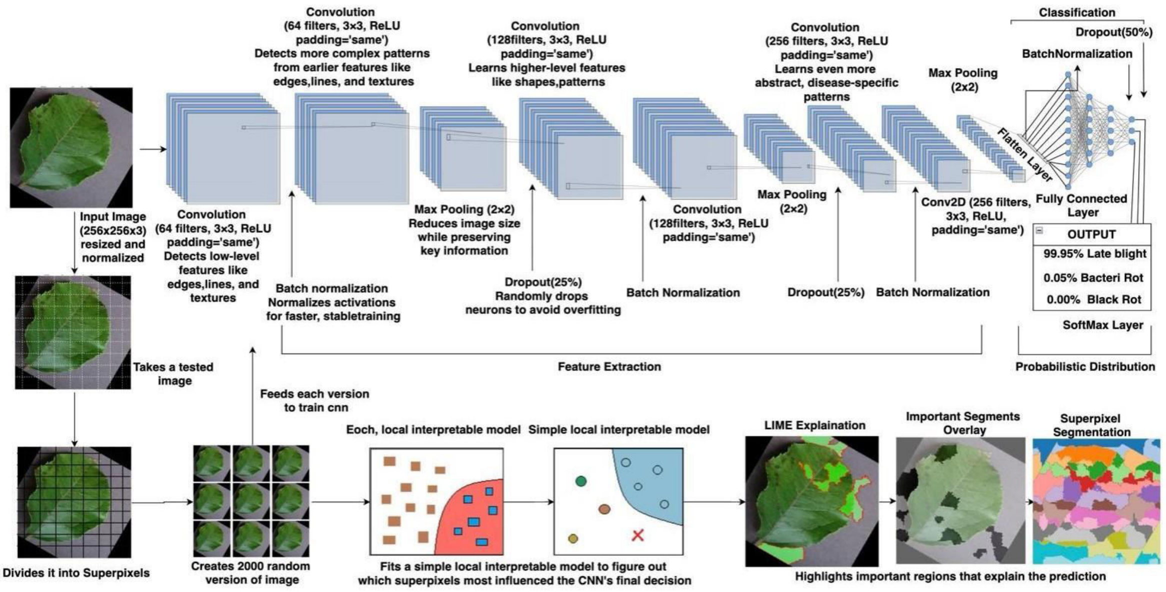


Fig. 4: Flowchart of CNN enhanced with LIME

E. Model Architecture

The computational framework employs a CNN designed for high-resolution plant disease classification. The architecture begins with three sequential blocks of convolutional layers (64, 128, and 256 filters) using 3×3 kernels, ReLU activation, and same padding to capture spatial hierarchies of disease patterns. Equation used (input tensor X and kernel W (with bias b)) [23]:

$$y^{k}_{i,j} = ReLU\left(\sum_{m,n} X_{i+m,j+n} \cdot W^{k}_{m,n} + b^{k}\right) \qquad (1)$$

k indexes the output channel (filter), ReLU(z)=max (0, z) is the activation function, "Same" padding ensures the output spatial dimensions remain the same as the input.

Each block incorporates batch normalization for stable training. After convolution, batch normalization is applied to stabilize and accelerate training. For a given activation x from the convolutional layer, batch normalization computes:

$$\hat{x} = \frac{x - \mu_{batch}}{\sqrt{\sigma^{2}_{batch} + \epsilon}} \qquad (2)$$

Max pooling (2×2) is employed to decrease the dimensions, and dropout layers (with a 25% rate) are included to avoid overfitting.

Dropout is applied to randomly deactivate a fraction p of the units during training, thereby reducing overfitting. Mathematically, this is represented as:

$$\tilde{x}_i = x_i \cdot d_i \qquad with \qquad d_i \sim Bernoulli(1 - p) \qquad (3)$$

where di is the binary mask that keeps a unit with probability 1−p [24].

A concluding convolutional block eliminates dropout to retain deeper feature representations. The network shifts to dense classification via a flattening layer, succeeded by a fully connected layer (512 units) that incorporates L2 regularization (λ=0. 001) and batch normalization, finishing with a 50% dropout layer for effective generalization. The output layer employs softmax activation across 38 nodes that relate to disease categories. This architecture balances feature extraction ability (through gradual filter expansion) with regularization techniques, optimized for distinguishing fine-grained visual symptoms in agricultural images while ensuring computational efficiency.

F. Interpretation of the model

Using quickshift superpixel segmentation (σ=3, max_dist=6), LIME explainability was applied to find important imaging biomarkers, producing quantitative feature contribution scores as well as visual explanations. For picture classification, the binary vector is a set that represents either 0 or 1. x ϵ {0,1} are coordinates that represent the pixels of a picture. For an example scenario, the LIME-obtained explanation is expressed as follows [25]:

$$\xi(x) = \arg min_{m} \left[U(p, m, Mx) + \omega(m) \right] \qquad (4)$$

Where 'x' represents sample and the interpretation is calculated by argmin$_{m}$ $U\ (p,\ m,\ Mx)$ and $\omega(m)$. 'p' represents the probability to a class, 'm' is defined as interpretable model, and to show the closer locality near *x*, a proximity measure M*x* is used. To measure the deceitfulness of the proposed model 'm' in approximating the probability value 'p' in the proximity defined by M*x the* parameter: $U\ (p,\ m,\ Mx)$ is used. The answer to how complex the model is, lies in $\omega(m)$. The explainer object of LIME (for sample, Blackbox model instance) consider using these steps for each of the samples: An image is selected (defined as illustrations) to analyse the blackbox prognosis; Creation of specimen dataset which has data focussed around a normal distribution; Samples that are nearest to the instance should be given more emphasis; Develop a model that is straightforward to comprehend using the sample dataset and its variants; Using the local model's interpretation, explain the prediction; and Describe what occurred in the example situation.

RESULTS AND DISCUSSION

In this section we discuss about the performance of the model architecture under the process of training and validation using the used Dataset. A thorough examination of the outputs of the disease prediction through LIME visual segmentation algorithms i.e., Quickshift algorithm and customized SLIC is given in this section.

A. *Training and Validation Performance:*

I. *Training Loss & Accuracy:* During the course of 10 epochs, the training loss consistently declined from approximately 1.6 to about 0.1, while the training accuracy increased from 55% to 95% (refer to Figure 5(a) and 5(b)).

II. *Validation Loss & Accuracy:* Following epoch 7, the validation loss stabilized at around 0.40, and accuracy peaked approximately 95.44%, which suggests strong generalization with only slight overfitting showing that the model is fit for detecting plant diseases.

Table 1 represents the performance of the customized model based on various performance metrices i.e., accuracy, precision, recall as shown Fig. 6.

Table 1: Performance of the Hybrid Model

Metric	Score
Accuracy	95.44%
Precision	95.59%
Recall	95.41%
F1 Score	95.42%

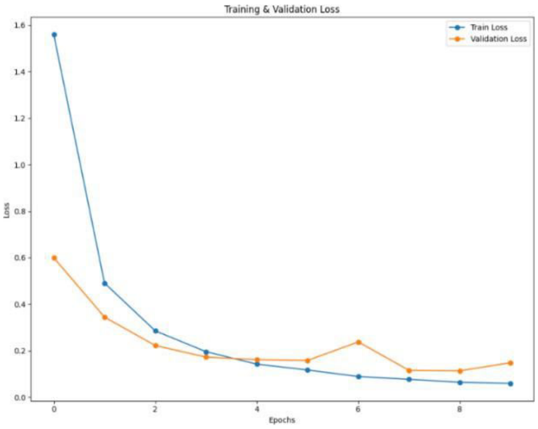


Fig. 5(a): Training and Validation Loss Curves

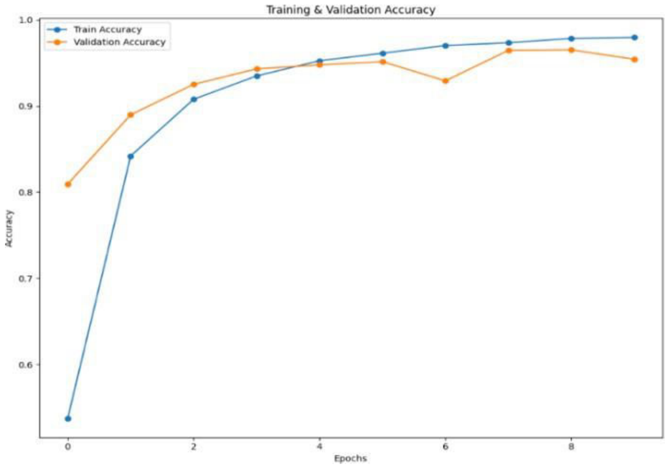


Fig. 5(b): Training and Validation Accuracy Curves

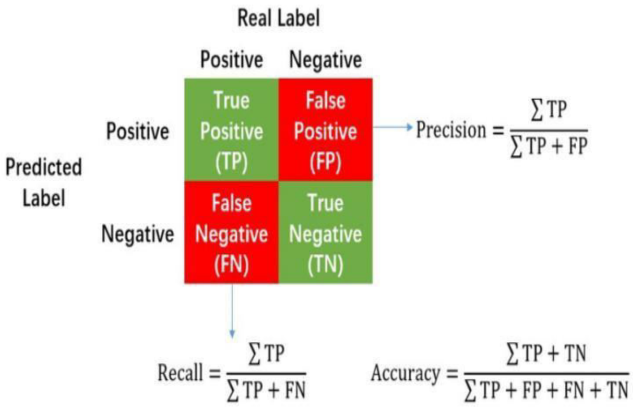


Fig. 6: Performance Metrics used for Model Evaluation

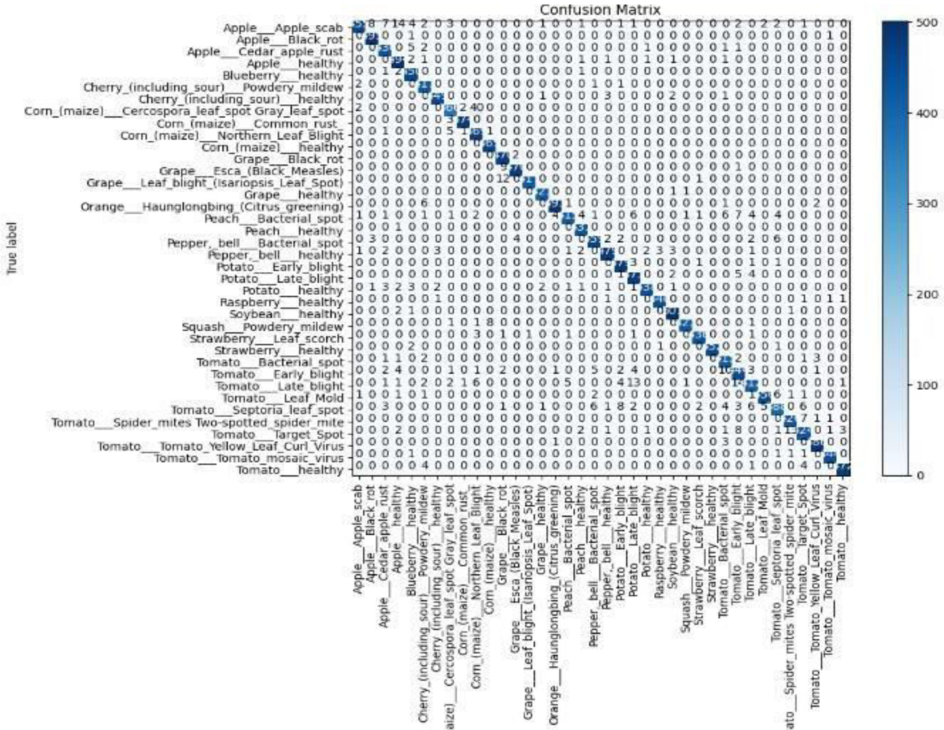


Fig. 7: Confusion Matrix for Hybrid Model

B. ***Confusion Matrix Analysis***

Key findings:

A multi-class image classifier's "confusion matrix" has been taught to identify 38 distinct plant-disease (and healthy-leaf) categories as given by the dataset used for training the model [26].

In this scenario (Fig. 7), the model is accurately identifying leaves in the majority of situations because mass is essentially on all of the diagonals. Here are some observations:

i.714 times, Applescab (cell @ row "Applescab" & column "Apple_scab") was properly classified.
ii.745 times, tomato_healthy was classified properly.

Configuration of the Matrix:

i.Rows = True labels (the true nature of the leaf)
ii.Columns = Labels Predicted (by the model's guess)
(becoming darker blue) and the number displayed.

From top-left to bottom-right, diagonal cells count how many predictions were right for each class. The number of times one class was mistaken for another is shown by cells that are not on the diagonal. As the count in each increase, so does the color intensity (becoming darker blue) and the number displayed. Table 2 represents the probability distribution of each class based on the feature segments.

Table 2: Probability Distribution

Class Name	**Probability**	**Bp Feature Segments explained**
Corn_(maize)___Common_rust_	100%	Key indicators in segments:67,30,83
Pepper,_bell___Bacterial_spot	0.0%	Key Indicators in segments:81,49,63
Tomato___Late_blight	0.0%	Key indicators in segments:119,96,18
Pepper,_bell___healthy	0.0%	Key indicators in segments:32,14,46
Apple___Cedar_apple_rust	0.0%	Key indicators in segments:24,5,94

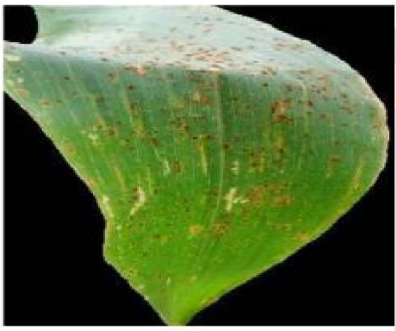


Fig. 8(a): Original Image

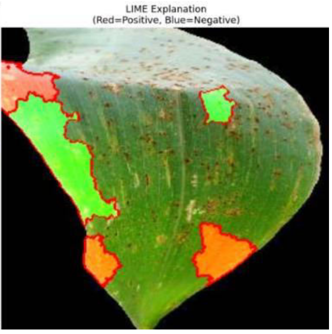


Fig. 8(c): Lime Explained Image

C. ***Visual LIME Overlays***

Quickshift Algorithm:

An example of an unsupervised image segmentation technique that generates "superpixels" based on the region's proximity and color (or feature) similarity is the Quickshift method [27]. It is critical that we understand the following ideas: Using clique intervals, an intervaling theory is constructed by combining all of the styles. Density Estimation Using Kernel Parzen Window and Clustering. Determine the location of each pixel in the following five-dimensional coordinate system: (x, y, r, g, b). Every dimensional frame has boundaries in terms of color and space.

Working of the Algorithm:

Kernel size manages the Gaussian kernel that is used to compute the local density for each pixel. Next, find the closest neighbor with a higher density that is located within max_dist in spatial space for each pixel. One super pixel is created from all of the pixels pointing to a high-density area. A segmentation map with approximately colored and spatially compact superpixel color homogeneity for each zone is the end result.

Visual Outputs:

The superpixel contribution map (Fig. 8(b)) of the original image (Fig. 8(a)) displays the top 10 superpixels colored according to their weight that had the greatest impact on the prediction for "common rust."

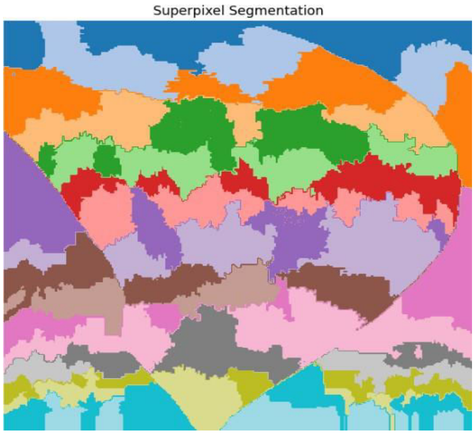


Fig. 8(b): Superpixel Image Segmentation

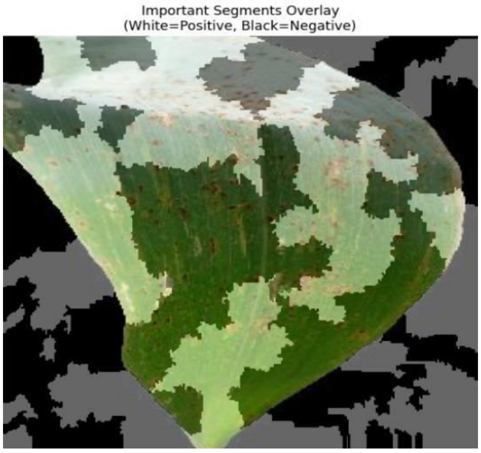


Fig. 8(d): Important Segments Overlay

The segmentation map (Fig. 8(d)) illustrates how the image was divided into different regions.

The overlay visualization (Fig. 8(c)) emphasizes areas contributing positively in red and those contributing negatively in blue.

SLIC Algorithm:

The SLIC algorithm (Simple Linear Iterative Clustering) is employed by LIME for image interpretation. LIME utilizes superpixels to segment the image into parts, while SLIC effectively creates these compact and uniform superpixels by grouping pixels based on color similarity and spatial closeness. This process helps LIME to generate a simplified version of the image for explanation, making it easier to understand how different characteristics of the input image impact a model's prediction [28].

Working of the Algorithm:

The image is over-segmented into what are known as superpixels, which are collections of nearby pixels with comparable hues or intensities. Instead of turning on and off individual pixels, LIME randomly activates and deactivates superpixels at the level of these clusters. For LIME, these subsequently turn into atomic perturbation units as shown in Fig. 9(a).

This significantly lowers the perturbation space's dimension (from about 100,000 pixels to about 50 superpixels), enabling LIME's sampling.

In reality, SLIC is five-dimensional k-means clustering. Color portions in Lab space (L, a, b) and pixel locations (x, y) by clustering in this coupled color-spatial region, SLIC ensures that each superpixel is both spatially compact and color-homogeneous.

Visual Outputs:

Segmentation Map for SLIC confirms your choice of n_segments, compactness, etc. visibly by displaying each superpixel in a distinct (tab20) colour (Fig. 9(b)).

Overlay of Positive Contributions Only (Fig. 9(c)) superpixels that increase the projected probability of the chosen class—that is, the segments that your model "depends on" the most—are highlighted.

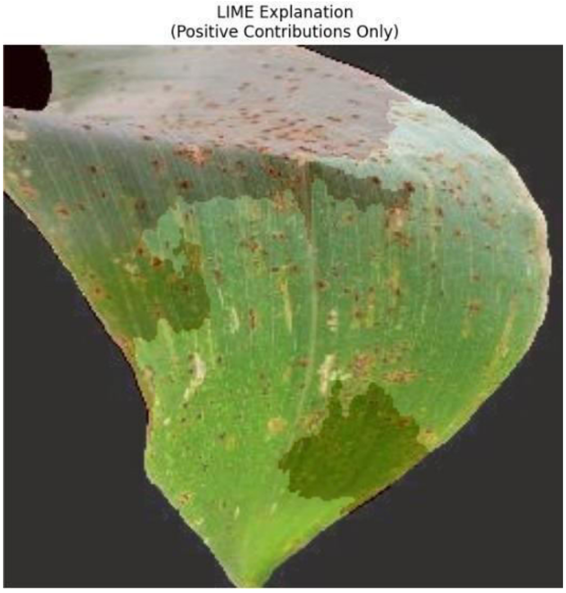
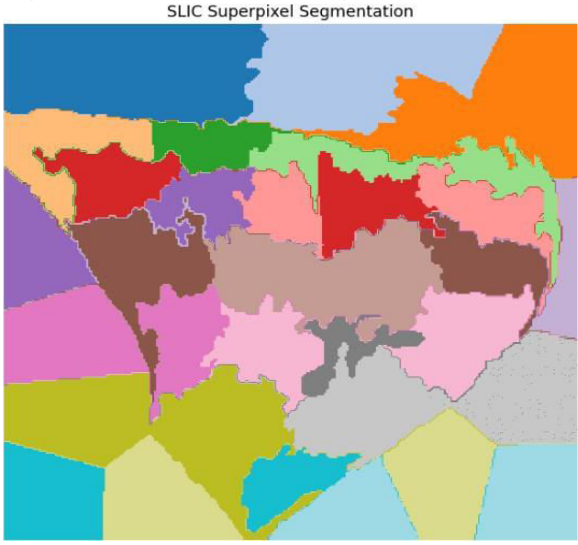


Fig. 9(a): Lime Explained Image



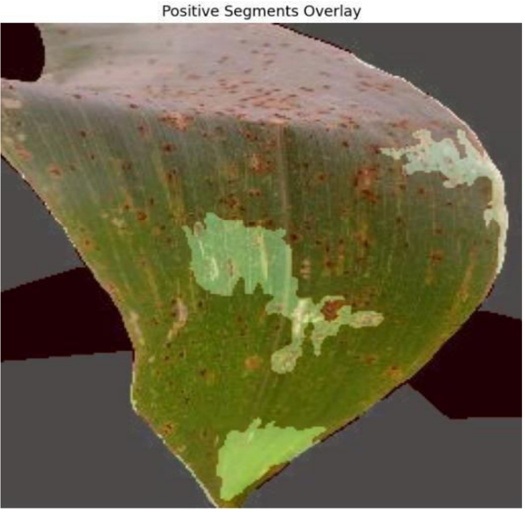


Fig. 9(b): SLIC Superpixel Segmentaion Fig. 9(c): Positive Segments Overlay

The below Table 3 represents the comparison of SLIC method with Quickshift method:

Table 3: Comparison of SLIC and Quickshift

Feature/Aspect	**SLIC**	**Quickshift**
Speed	Faster, highly efficient	Moderate (slower than SLIC)
Edge Adherence	Very Good (follows object boundaries)	May ignore fine edges depending on compactness

Type	Graph based, model seeking	Cluster based
Suitability for LIME	More common choice	Possible, but less predictable segment size

I. Conclusion

Though many automated and computer vision methods exist for detecting and categorizing plant diseases, significant gaps remain in this research area. In this research, we employed an enhanced data which had huge variety of healthy and diseased plant leaves. This study helped in developing a CNN based model which is used for multi-class plant disease categorization covering 38 categories using an augmented dataset of leaf images. The model achieved an overall test accuracy of 97.96% with a balanced accuracy of 95.44%, precision of 95.59% and recall of 95.41%, and an F1 score of 95.42%. The difference between the training accuracy of 95% and validation/test performance of roughly 90% and 89% respectively illustrates good generalization with minimal overfitting, primarily attributed to the data augmentation techniques used. Looking at the confusion matrix, it was revealed that most misclassification errors came from visually identical disease pairs such as tomato early blight versus late blight, which highlights the problem of using appearance-based diagnosis.The use of LIME explainability confirmed that the CNN's decision-making process adheres to agronomic standards. Superpixels containing distinct lesions had the most substantial positive impact, while healthy areas displayed negative weights. This transparency enhances trust in automated diagnostics by demonstrating the model's focus on biologically relevant features over irrelevant background elements. CNN-based methods can offer a strong basis for plant disease screening systems by combining excellent predicted accuracy with clear and understandable explanations when paired with local interpretability methodologies.To address the real-world implementation feasibility and reliability of plant disease classification systems, future research should address key challenges such as class imbalance, environmental variability, and model explainability. Implementation strategies may include using class-weighted loss functions, incorporating diverse image conditions, leveraging high-resolution and multi-modal inputs, and improving segmentation fidelity through advanced superpixel algorithms. Explainability can be strengthened using global interpretation tools like SHAP and model deployment on edge devices can be supported through compression techniques. Field trials and user studies will be essential to validate model effectiveness, improve user trust, and improve human-AI interaction for practical, scalable disease detection solutions in agriculture.

REFERENCES

[1] Alam, T., Jowthi, C. B., & Pathak, A. (2024). Comparing pre-trained models for efficient leaf disease detection: a study on custom CNN. Journal of Electrical Systems and Information Technology, 11(1), Article 12.

[2] Schallner, J. Rabold, O. Scholz, and U. Schmid, "Effect of Superpixel Aggregation on Explanations in LIME – A Case Study with Biological Data," *arXiv preprint arXiv:1910.07856*, 2019.

[3] W. Hryniewska, A. Grudzień, and P. Biecek, "LIMEcraft: handcrafted superpixel selection and inspection for visual explanations," *Machine Learning*, vol. 113, pp. 3143–3160, 2022, doi: 10.1007/s10994-022-06204-w.

[4] P. Gohel, P. Singh, and M. Mohanty, "Explainable AI: current status and future directions," *IEEE Access*, vol. 5, pp. 52138-52160, 2017, doi: 10.1109/ACCESS.2017.DOI.

[5] V. Boppana, S. Davuluri, R. Reddi, N. Syda, M. U. Rao, A. Kasukurthi, R. Badiguntla, and Y. N. Rao, "Plant disease detection using hybrid MobileNetV2-compact CNN architecture with LIME integration," *Journal of Information Systems Engineering and Management*, vol. 10, no. 13s, pp. 1–12, 2025.

[6] Amrita S. Tulshan and Nataasha Raul, "Plant Leaf Disease Detection using Machine Learning," 10th ICCCNT 2019, July 6-8, 2019, IIT - Kanpur, Kanpur, India. DOI: 10.1109/ICCCNT45670.2019.8944556.

[7] Govardhan, M., & Veena, M. B. (2019). Diagnosis of Tomato Plant Diseases using Random Forest. In 2019 Global Conference for Advancement in Technology (GCAT) (pp. 1-5). IEEE. DOI: 978-1-7281-3694-3.

[8] Hasan, M. et al., "Maize Diseases Image Identification and Classification by Combining CNN with Bi-Directional Long Short-Term Memory Model," in 2020 IEEE Region 10 Symposium (TENSYMP), Dhaka, Bangladesh, 5-7 June 2020.

[9] Santha Hari, S., Karthikeyan, S., Sivakumar, M., Suriya, S., & Renuga, P. (2019). Detection of Plant Disease by Leaf Image Using Convolutional Neural Network. In 2019 International Conference on Vision Towards Emerging Trends in Communication and Networking (ViTECoN) (pp. 1-5). IEEE. doi: 10.1109/VITECoN.2019.8899514.

[10] D. Jayakumar, A. Elakkiya, R. Rajmohan, M. O. Ramkumar, "Automatic Prediction and Classification of Diseases in Melons using Stacked RNN based Deep Learning Model," 2020.

[11] S. Saha and S. M. Ahsan, "Rice Disease Detection using Intensity Moments and Random Forest," 2021 International Conference on Information and Communication Technology for Sustainable Development (ICICT4SD), Dhaka, Bangladesh, 2021, pp. 165-170, doi: 10.1109/ICICT4SD50815.2021.9396986.

[12] Sirohi, A., & Malik, A. (2021). A Hybrid Model for the Classification of Sunflower Diseases Using Deep Learning. In 2021 2nd International Conference on Intelligent Engineering and Management (ICIEM) (pp. 58-62). IEEE. DOI: 10.1109/ICIEM52530.2021.00017.

[13] Saurav Sagar, Mohammed Javed, and David S. Doermann. "Leaf-Based Plant Disease Detection and Explainable AI." (2024).

[14] Kim, S. et al. (2023). A novel deep learning-based disease detection model for Solanaceae crops using leaf images. Scientific Reports, 13, 7331. https://doi.org/10.1038/s41598-023-34549-2 .

[15] Chopda, J., Nakum, S., & Nakrani, V. (2023). Cotton Crop Disease Detection using Decision Tree Classifier. Universal College of Engineering.

[16] Anwar Rifa'i and Deni Mahdiana, "Image Processing for Diagnosis Rice Plant Diseases Using the Fuzzy System," 2020 International Conference on Computer Science and Its Application in Agriculture (ICOSICA),2020,pp.1-5.DOI: 10.1109/ICOSICA49951.2020.9243274.

[17] Islam, M., Dinh, A., Wahid, K., & Bhowmik, P. (2017). Detection of Potato Diseases Using Image Segmentation and Multiclass Support Vector Machine. In 2017 IEEE 30th Canadian Conference on Electrical and Computer Engineering (CCECE) (pp. 1-5). IEEE.

[18] Rajesh, B., Vardhan, M. V. S., & Sujihelen, L. (2020). Leaf Disease Detection and Classification by Decision Tree. In Proceedings of the Fourth International Conference on Trends in Electronics and Informatics (ICOEI 2020). IEEE Xplore Part Number: CFP20J32-ART; ISBN: 978-1-7281-5518-0.

[19] Pranjali B. Padol and Prof. Anjali A. Yadav, "SVM Classifier Based Grape Leaf Disease Detection," 2016 Conference on Advances in Signal Processing (CASP), Cummins College of Engineering for Women, Pune, Jun 9-11, 2016. DOI: 978-1-5090-0849-0/16/$31.00 ©2016 IEEE.

[20] Bhowmik, S., Talukdar, A. K., & Sarma, K. K. (2020). Detection of Disease in Tea Leaves Using Convolution Neural Network. In 2020 Advanced Communication Technologies and Signal Processing (ACTS) (pp. 1-6). IEEE. DOI: 10.1109/ACTS49415.2020.9350413.

[21] MohamadAzmi, Muhammad Thaqif bin, and Naimah Mat Isa. "Orchid Disease Detection Using Image Processing and Fuzzy Logic." 2013 International Conference on Electrical, Electronics and System Engineering. IEEE, 2013.

[22] Hossain, E., Hossain, M. F., & Rahaman, M. A. (2019). A Color and Texture Based Approach for the Detection and Classification of Plant Leaf Disease Using KNN Classifier. 2019 International Conference on Electrical, Computer and Communication Engineering (ECCE), 7-9 February, 2019. DOI: 10.1109/ECACE.2019.8679247.

[23] H. Raval and J. Chaki, "Ensemble transfer learning meets explainable AI: A deep learning approach for leaf disease detection," *Ecological Informatics*, vol. 84, p. 102925, 2024.

[24] Ethiraj, R. P., & Paranjothi, K. (2024). A deep learning-based approach for early detection of disease in sugarcane plants: an explainable artificial intelligence model. International Journal of Artificial Intelligence, 13(1), 974-983. DOI: 10.11591/ijai.v13.i1.pp974-983.

[25] S. Sladojevic, M. Arsenovic, A. Anderla, D. Culibrk, and D. Stefanovic, "Deep Neural Networks Based Recognition of Plant Diseases by Leaf Image Classification," *BioMed Research International*, vol. 2016, Article ID 3285901, 2016. DOI: 10.1155/2016/3285901

[26] R. H. Prince, A. A. Mamun, H. I. Peyal, S. Miraz, M. Nahiduzzaman, A. Khandakar, and M. A. Ayari, "CSXAI: a lightweight 2D CNN-SVM model for detection and classification of various crop diseases with explainable AI visualization," *Frontiers in Plant Science*, vol. 15, p. 1412988, Jul. 2024. doi: 10.3389/fpls.2024.1412988.

[27] P. Patil, S. Pamali, S. Devagiri, and S. J. Mirje, "Plant leaf disease detection using XAI," *IEEE Access*, vol. 12, pp. 1–9, 2024.

[28] Z. Zainab and R. Mahum, "Plant disease detection using deep learning techniques," *IECE Journal of Image Analysis and Processing*, vol. 1, no. 1, pp. 45–53, 2025.