RESEARCH PAPER

Deep Learning-Based Disease Detection in Grape Leaves and Pomegranate Fruits Using CNN.

Manasi Kulkarni [1], Dr.Smita Desai[2]

[1]*PG Student, Electronics and Telecommunication Dr.D.Y.Patil Institute of Technology, Pimpri Pune, India*
manasi2731987@gmail.com
[2]*Associate Professor,Electronics and Telecommunication Dr.D.Y.Patil Institute of Technology, Pimpri Pune, India*
smita.desai@dypvp.edu.in

ABSTRACT

Plant diseases pose a major threat to global agricultural productivity, causing significant yield losses and compromising fruit quality. Grape and pomegranate crops, in particular, are highly vulnerable to fungal, bacterial, and environmental stresses, which manifest as visible symptoms on leaves and fruits. Traditional disease detection methods rely heavily on manual inspection, expert evaluation, and laboratory testing, which are labor-intensive, time-consuming, and subjective. Recent advancements in artificial intelligence (AI), specifically deep learning, offer powerful alternatives through automated image-based disease diagnosis. Convolutional Neural Networks (CNNs) have demonstrated exceptional capability in extracting hierarchical visual features, detecting subtle lesion patterns, and achieving high-accuracy classification even under complex environmental conditions. This research presents a deep learning–based framework for automated disease detection in grape leaves and pomegranate fruits. The system utilizes optimized CNN architectures designed to extract discriminative features from heterogeneous image datasets, including early blight, downy mildew, powdery mildew, bacterial spots, fruit cracks, wilt, heart rot, and sunburn. The proposed framework integrates preprocessing, augmentation, multi-class classification, and hyperparameter tuning to ensure robust generalization under natural field settings. Performance evaluation includes accuracy, precision, recall, F1-score, confusion matrices, and cross-validation. Experimental results demonstrate that CNN-based systems outperform traditional image-processing techniques and offer reliable early detection capabilities. The study highlights the potential of AI-driven tools to reduce crop losses, optimize pesticide use, enhance decision-making, and support precision agriculture practices. This work contributes a scalable, efficient, and real-time disease detection solution for grape and pomegranate cultivation

Keywords: *Deep Learning, Convolutional Neural Networks, Plant Disease Detection, Grape Leaves, Pomegranate Fruits*
How to cite this article: Kulkarni M, Desai S, Deep Learning-Based Disease Detection in Grape Leaves and Pomegranate Fruits Using CNN.. .Int J Drug Deliv Technol. 2026;16(1s): 1111-1119;DOI: 10.25258/ijddt.16. 1111-1119
Source of support: Nil.
Conflict of interest: None

INTRODUCTION

Agriculture forms the backbone of food security and economic stability in most developing countries. As the global population continues to rise, there is an increasing demand for high-quality agricultural produce, especially fruits that offer nutritional and commercial value. Among them, grapes and pomegranates stand out due to their medicinal, economic, and industrial significance. Grapes constitute one of the world's most traded fruit crops, widely utilized in fresh consumption, winemaking, juice production, and food processing industries. Similarly, pomegranates are valued for their antioxidant properties, high vitamin content, and growing demand in international markets. However, both crops are significantly affected by a wide spectrum of diseases that hinder productivity, degrade fruit quality, and cause substantial financial losses to farmers.

1.1 Challenges in Manual Disease Detection

Traditional disease detection in agriculture primarily relies on manual examination by farmers or plant pathologists. This approach poses several challenges:

Subjectivity: Disease severity assessments often vary from person to person, leading to inconsistent results.

Time-consuming: Large-scale farms make it difficult to inspect every plant regularly.

Lack of expertise: Small and marginal farmers may not have access to trained agronomists.

Late detection: Many diseases progress rapidly, and early symptoms may go unnoticed.

Environmental variability: Lighting conditions, leaf orientation, and background noise complicate visual evaluation.

Given these limitations, there is a pressing need for an automated, reliable, and scalable solution that can identify diseases early and accurately.

1.2 Role of Artificial Intelligence in Agriculture

The integration of Artificial Intelligence (AI) and Computer Vision has revolutionized modern agriculture. Deep Learning, particularly Convolutional Neural Networks

---

**Author for Correspondence: Manasi Kulkarni*

(CNNs), has emerged as the most powerful approach for image-based plant disease detection. CNNs learn hierarchical patterns directly from raw images, eliminating the need for handcrafted features traditionally used in image processing methods.

Unlike classical techniques that depend on manually extracted features—such as color histograms, texture descriptors, or shape patterns—CNNs automatically learn low-level features (edges, textures) and high-level representations (lesion shape, discoloration, infection spread). This enables robust detection under diverse illumination, orientation, and background conditions frequently encountered in field environments.

1.3 Diseases Affecting Grapes and Pomegranates

Grape and pomegranate crops are highly vulnerable to a variety of fungal, bacterial, viral, and abiotic disorders. Some major diseases include:

Grape Diseases: Downy mildew, Powdery mildew, Black rot, Anthracnose, Leaf blight, Bacterial spot

Pomegranate Diseases: Fruit cracking, Bacterial blight, Wilt, Heart rot, Cercospora leaf spot, Sunburn, Alternaria fruit rot

Early identification of these diseases is critical for preventing widespread damage, reducing pesticide consumption, and maintaining crop health.

1.4 Importance of Automated Disease Detection

AI-based disease detection systems have the potential to transform fruit and leaf analysis in real-world agricultural settings. Some benefits include:

Precision Agriculture: Offers targeted pesticide application, reducing environmental impact.

Cost Efficiency: Prevents excessive use of chemicals and minimizes yield losses.

Real-time Diagnosis: Supports smartphone-based field applications for instant results.

Scalability: Can be deployed in large orchards where manual monitoring is difficult.

Decision Support: Helps farmers take corrective actions early, improving yield quality.

1.5 Deep Learning for Grape and Pomegranate Disease Detection

Recent studies show that CNN-based systems consistently outperform traditional approaches due to:

Robust feature extraction
High classification accuracy
Applicability across crops and disease types
Ability to generalize from diverse datasets

Deep learning models such as VGG16, ResNet50, MobileNet, EfficientNet, and custom CNN architectures have achieved outstanding results for plant disease classification tasks. For grape and pomegranate disease detection, CNNs analyze leaf and fruit images to identify lesions such as:

White powdery textures (powdery mildew)
Yellowish oil spots (downy mildew)
Blackened lesions or rot patches
Cracks and deformities (pomegranate fruit cracking)

These visual symptoms contain unique patterns that CNNs excel at learning.

1.6 Dataset Importance and Challenges

Developing a robust deep learning model requires large, diverse, and well-labeled datasets. However, agricultural datasets often suffer from:

Limited availability (rare diseases are underrepresented)
Class imbalance (healthy leaves dominate datasets)
Environmental variations (lighting, background clutter)
Noise and distortions (shadows, dust, occlusions)

To address these challenges, preprocessing steps such as image resizing, normalization, and noise reduction are essential. Data augmentation—including rotation, flipping, zooming, contrast enhancement, and color jittering—helps increase dataset diversity and improves model generalization.

1.7 Significance of the Proposed Research

This research aims to develop a CNN-based integrated framework capable of detecting diseases in both grape leaves and pomegranate fruits. The significance of the proposed system includes:

Providing a unified model applicable across multiple crops
Ensuring robust detection in field conditions
Supporting farmers through easy-to-use mobile and web interfaces
Contributing to sustainable farming practices

LITERATURE REVIEW

Deep learning has emerged as the dominant paradigm for image-based plant disease detection because of its ability to automatically extract hierarchical features from complex and variable agricultural imagery. A foundational study by Pacal et al. [1] conducted a systematic review of deep learning strategies applied across diverse plant species and disease categories. Their work established that convolutional neural networks (CNNs) consistently outperform classical machine learning approaches due to robustness against illumination variations, background clutter, and biological variations in leaves and fruits. They highlighted the superiority of architectures such as VGG, Inception, ResNet, and DenseNet for agricultural classification tasks, while identifying dataset imbalance and limited field-realistic samples as persistent challenges.

For grape disease detection, Kunduracioglu and Pacal [2] presented a comprehensive analysis of deep learning advancements tailored specifically to grape leaves. They emphasized recent architectural improvements, hybrid CNN models, and fine-tuning techniques that significantly improved accuracy in identifying downy mildew, powdery mildew, black rot, and other common grapevine diseases. Their work demonstrated that CNN-based approaches can achieve expert-level accuracy when supported by high-quality annotated datasets.

Lightweight architectures have also gained prominence due to their suitability for mobile deployment and edge computing in precision agriculture. Karim et al. [3] proposed a compact CNN architecture integrated with Grad-CAM to classify grape leaf diseases in real-time on edge devices. Their model achieved high accuracy while maintaining low computational complexity, making it suitable for use in handheld devices and field-based monitoring systems. Similarly, Zhang et al. [4] developed an automatic deep learning–driven system for grape downy mildew detection, achieving promising recall and precision metrics. Their research showed that CNNs excel in identifying early symptoms such as oil spots and chlorotic patches.

Pomegranate disease detection, though comparatively less explored than grape disease detection, has seen significant contributions in recent years. Sahebgouda et al. [5] introduced a CNN-LSTM hybrid framework that improved pomegranate disease classification by integrating temporal features, demonstrating benefits in cases where disease progression patterns are relevant. In another innovative contribution, Sajitha et al. [10] developed a hybrid optimal attention capsule network for pomegranate fruit disease diagnosis. Their approach integrated attention mechanisms with capsule networks to improve feature localization and class discrimination, yielding superior performance in multi-class settings.

General advancements in deep learning for plant pathology have also contributed significantly to grape and pomegranate disease detection. Ashurov et al. [6] proposed a depthwise CNN enriched with squeeze-and-excitation (SE) blocks and residual skip connections, demonstrating improved feature extraction efficiency and classification accuracy. Their research supports the trend toward architectural modularization and efficient feature recalibration techniques. Ishengoma and Lyimo [7] explored ensemble learning, combining multiple CNN feature extractors with a Random Forest classifier for grape leaf disease detection. Their ensemble model improved classification accuracy and robustness, especially for visually similar disease classes.

The broader research community has also highlighted the growing importance of deep learning in pest and disease monitoring. Wang et al. [9] presented a comprehensive review on deep learning applications for plant disease and pest identification using remote sensing data. They emphasized the increasing use of multispectral and hyperspectral imaging, attention mechanisms, and transformer-based architectures in complex agricultural environments. Leite et al. [11] focused on CNN-based severity estimation—a crucial extension beyond mere detection. Their review emphasized the need for quantitative assessment frameworks capable of grading infection severity, enabling better decision-making for pesticide application.

IoT-assisted plant health monitoring systems have also been incorporated into disease detection frameworks. Mulani et al. [12] proposed an IoT-based air, water, and soil monitoring system for pomegranate farming, demonstrating the importance of integrating environmental sensing with disease detection for holistic crop management. Kashid and Karande [13] also contributed to IoT-based environmental monitoring using machine learning, reinforcing the role of contextual variables—such as temperature, humidity, and soil moisture—in disease onset prediction.

Deep learning applications in related domains also provide valuable insights into model optimization and interpretability. Mulani et al. [14] discussed the role of deep learning in personalized medicine, emphasizing model generalizability, data-driven decision-making, and interpretability—concepts directly applicable to agricultural disease detection scenarios. Karve et al. [15] proposed an optimized neural network for neurological disorder prediction, highlighting the benefits of hyperparameter tuning and architectural optimization, which are also essential for improving agricultural CNN models.

Desai *et al.* [16] proposed a novel crop disease detection approach that emphasizes efficient feature extraction to enhance classification performance while reducing computational overhead. Their method combines carefully designed feature descriptors with machine learning classifiers to achieve improved detection accuracy across multiple crop types. The study demonstrated that optimized feature extraction significantly enhances robustness against variations in illumination, leaf orientation, and background complexity. However, the approach still relies on traditional machine learning pipelines, which may limit scalability when dealing with large-scale datasets compared to deep learning-based models.

In earlier work, Desai and Kanphade [17] investigated the role of classical image processing techniques in leaf disease identification, with a particular focus on median filtering for noise reduction. Their study highlighted that effective preprocessing is a critical step in improving the visibility of disease symptoms such as spots, discoloration, and texture irregularities. By applying median filtering prior to feature extraction, the authors achieved improved segmentation and classification accuracy. While the method is computationally simple and suitable for low-resource systems, it is sensitive to complex field conditions and requires manual feature design, which can limit adaptability across diverse crops and disease types.

Overall, the reviewed literature reveals strong evidence that CNN-based systems are highly effective for disease detection in both grape leaves and pomegranate fruits. The integration of SE blocks, attention mechanisms, transfer learning, Grad-CAM explainability, ensemble learning, and edge-device optimization has collectively contributed to the advancement of reliable and scalable solutions. However, key challenges persist—such as limited datasets, class imbalance, environmental variability, and low generalization to real-field conditions. Addressing these limitations remains central to future research. The existing body of work demonstrates that deep learning, particularly CNN-based architectures, has transformed agricultural disease detection by offering high accuracy, automation, and robustness. Researchers have developed optimized CNN architectures, lightweight models for deployment on edge devices, ensemble frameworks for improved classification, and attention-enabled networks for enhanced feature extraction. While grape disease detection has received more extensive attention, emerging studies on pomegranate disease detection—especially those involving hybrid and attention-based models—show promising advancements. IoT-driven environmental monitoring provides additional context for predictive disease modeling. Despite these advancements, challenges such as dataset limitations, environmental variability, and the need for severity estimation frameworks remain open, offering opportunities for further innovation in deep learning–based plant disease detection.

METHODOLOGY

Grapes and Pomegranate Disease Detection using CNN (Mathematical Model)

In a CNN model, the input data is passed through a series of layers that are designed to extract increasingly abstract features and then do the classification. The basic building blocks of a CNN are convolutional layers, which use filters to extract features from the input data, and pooling layers, which down sample the output of the convolutional layers to reduce the dimensionality of the data. After passing through several convolutional and pooling layers, the output is then flattened and fed into a series of fully connected layers, which perform classification or regression on the extracted features.

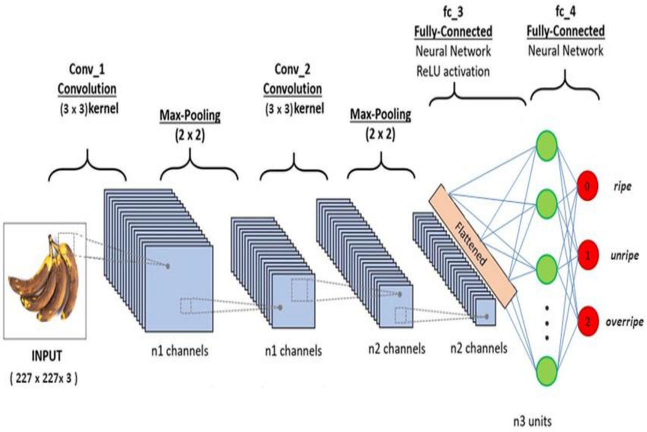


Figure 1: CNN Model Architecture

The CNN-model has following two parts:

Feature Learning

Classification

1. Feature Learning

The feature learning involves the techniques that are used to extract the features from input images and make the machine to learn those features automatically. In CNN, the features are extracted through two components: Convolution Layer and Pooling Layer. Let us describe the two layers one by one :

Convolution Layer

The convolution layer is basically used for the feature extraction. It does the feature extraction by firstly applying convolution function and then activation function on the output of convolution function. There are multiple numbers of convolution layers which are used for the feature extraction.

In the convolution operation, we use a linear function known as the kernel function to extract the features. This kernel function is also known as the filter.

Suppose we have an input image described by tensor **I** of dimension m1 x m2 x mc. where:

$$m_1 = height\ of\ image$$
$$m_2 = width\ of\ image$$
$$m_c = number\ of\ channels$$

We apply a filter which is also a tensor of dimension (n1 x n2 x nc). (the number of channels for the kernel is the same as the input image). The filter moves upon the image from left to right ,do multiplication between that part of I and K and sum up those products. Stride determines the step size by which the filter would move upon image. The resultant of I and K is another tensor of dimension (m1 — n1 +1) x (m2 — n2 +1) x 1 .

Here,

$$\dim of\ I = m_1 \times m_2 \times m_c ------(1)$$
$$\dim of\ K = n_1 \times n_2 \times n_c ----(2)$$
$$\dim of\ F = (m_1 - n_1 + 1) \times (m_2 - n_2 + 1) \times 1 ----(3)$$

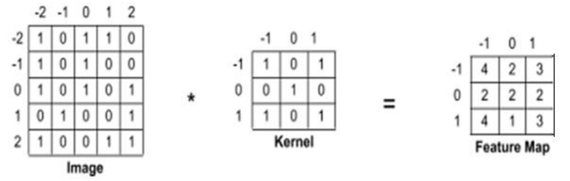
And,

$$F[i,j] = (I * K)_{[i,j]} ----(4)$$

The ij-th entry of the feature map is given as below:

$$f[i,j] = \sum_{x}^{m_1} \sum_{y}^{m_2} \sum_{z}^{m_c} K_{[x,y,z]} I_{[i+x-1,j+y-1,z]} -------(5)$$

We have taken the following example of a 5x5x1 dimensional image being convoluted with a kernel of 3x3x1 and the stride s= 1 has been used.



	-2	-1	0	1	2
-2	1	0	1	1	0
-1	1	0	1	0	0
0	1	0	1	0	1
1	0	1	0	0	1
2	1	0	0	1	1

Image

*

	-1	0	1
-1	1	0	1
0	0	1	0
1	1	0	1

Kernel

=

	-1	0	1
-1	4	2	3
0	2	2	2
1	4	1	3

Feature Map

The ij-th entry of feature map is given by following general formula in case of single channel:

$$f[i,j] = (I * K)_{[i,j]} \sum_{x}^{m_1} \sum_{y}^{m_2} K_{[x,y]} I_{[i-x,j-y]} ----(6)$$

Let us compute the 11-entry of the feature map in above example:

$$f[i,j] = \sum_{x=1}^{m_1} \sum_{y=1}^{m_2} \sum_{z=1}^{m_c} K_{[x,y,z]} I_{[i+x-1,j+y-1]} -----(7)$$

$$f[1{,}1] = \sum_{x=-2}^{5}\sum_{y=-2}^{5} K_{[x,y]} I_{[1-x,1-y]}$$

$$= \sum_{x=-2}^{5} K_{[x,-2]} I_{[1-x,1]} + \sum_{x=-2}^{5} K_{[x,-1]} I_{[1-x,2]} +$$

$$\sum_{x=-2}^{5} K_{[x,0]} I_{[1-x,1]} + \sum_{x=-2}^{5} K_{[x,1]} I_{[1-x,0]} + \sum_{x=-2}^{5} K_{[x,2]} I_{[1-x,-1]}$$

f[1, 1] = & K[-2, -2]I[3, 3] + K[-1, -2]I[2, 3] + K[0, -2]I[1, 3] + K[1, -2]I[0, 3] + \\ & K[2, -2]I[-1, 3] + K[-2, -1]I[3, 2] + K[-1, -1]I[2, 2] + K[0, -1]I[1, 2] + K[1, -1]I[0, 2] + \\ & K[2, -1]I[-1, 2] + K[-2, 0]I[3, 1] + K[-1, 0]I[2, 1] + K[0, 0]I[1, 1] + K[1, 0]I[0, 1] + \\ & K[2, 0]I[-1, 1] + K[-2, 1]I[3, 0] + K[-1, 1]I[2, 0] + K[0, 1]I[1, 0] + K[1, 1]I[0, 0] + \\ & K[2, 1]I[-1, 0] + K[-2, 2]I[3, -1] + K[-1, 2]I[2, -1] + K[0, 2]I[1, -1] + K[1, 2]I[0, -1] + \\ & K[2, 2]I[-1, -1]
The entries which are not available are substituted zero.

f[1, 1] = & 0 + 0 + 0 + 0 + 0 + 0 + (1 \times 1) + (0 \times 1) + (1 \times 0) + (0 \times 1) + 0 + 0 + (1 \times 0) + \\
& 0 + 0 + 0 + 0 + (1 \times 1) + 0 + 0 + (1 \times 1) + 0 + 0 + 0 + 0 + 0 + 0
Similarly, the remaining entries can be calculated by using the formula.
The whole procedure is repeated by applying different types of filters which collect different features of image like blur, sharpness etc. Here , the number of filters could be more than one that defines the concept of stride.

Padding

The procedure described above has one drawback which the applied filters focus more upon the center of image rather than its corner. It could be compensated by padding. By convention , zero padding is in practice in which a column and row of zeros has been added on each side of the input tensor.

Activation Function

In general , one bias term 'b' has been added to the convoluted part and then the activation function is applied.

$$c = F + b - - - - (8)$$

$$c = I * K + b - - - (9)$$

$$Conv(I, K) = \emptyset_a(c) = \emptyset_a(I * K + b)$$

$$where\ \emptyset_a\ is\ an\ activation\ function$$

There are different types of activation functions as sigmoid, tangent, hyperbolic tangent function. The most commonly used activation function is ReLU which eliminates the negative values:

$$R(x) = \max(0, x) - - - - - (10)$$

Press enter or click to view image in full size

2. Pooling Layer

In the pooling layer ,the spatial size of convoluted features is reduced. In this way we get the dominant feature of the image. In the pooling layer, a pooling function is applied on the result obtained from the convolution layer. Let us assume that:

$$Conv(I, K) = C - - - - (11)$$

$$P = \emptyset_p(C)\ where\ \emptyset_p is\ an\ pooling\ function$$

The dimension of pooled part is given as:

$$\dim of\ P = \left(\frac{m_1 + 2_p - n_1}{s}\right) \times \left(\frac{m_2 + 2_p - n_2}{s}\right) \times m_c - - - - (12)$$

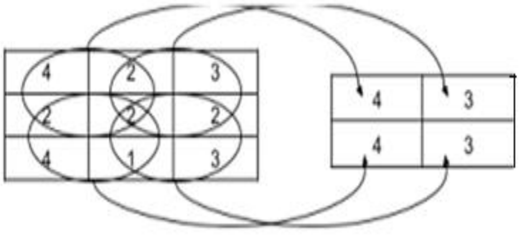
$$where, m_1 \times m_2 = the\ dimensions\ of\ input\ image$$

$$n_1 \times n_2 = the\ dimensions\ of\ padding\ kernel$$

$$here, Here'sstands\ for\ stride\ and\ p\ stands\ for\ padding.$$

There are different types of pooling such as sum pooling, average pooling, max pooling. Example of max pooling is given below. The maximum pooling is done upon 2x2 patches. From each patch, the maximum entry is selected.

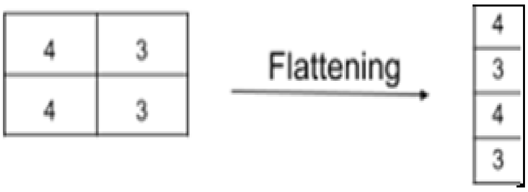
Press enter or click to view image in full size



Max Pooling

2. Classification

In this way, a number of hidden layers(convolutional layer + pooling layer) has been used for the feature extractions. Once the feature extraction is done, the resultant is flattened into a single vector. Now, this single vector would be used as the input for the fully connected layer where the classification is done.



Fully Connected Layer

The fully connected layer receives the flattened vector and results into another vector. In machine learning models, there is also the possibility of one class to occur higher as compared to other class. So, to eliminate this problem the balanced weights are combined with the pooled part, a biased term has been added and then the activation function is applied.

The mathematical description is as below:

$$X = \sum_i w_i P_i + b' - - - - (13)$$

$$z = g(X) - - - - (14)$$

$Where\ g\ is\ an\ activation\ function\ for\ the\ fully\ connected\ layer$

In this way, on each layer the weights are added to the pooled parts and activation function is applied. A number of hidden layers are used here and the last layer would use the activation function which performs the classification by calculating the probability for each class.

The summary has been described as below:

Input image is passed through a series of convolutional layers that extract features using filters.

Output of each layer is passed through a rectified linear unit (ReLU) activation function to introduce non-linearity.

Pooling layers are used to reduce dimensionality of output and make the model more efficient.

The convolutional and pooling layers can be repeated several times, depending on the complexity of the problem.

Output of the final convolutional layer is flattened and fed into a fully connected layer (FC Layer) that performs classification or regression on the extracted features.

The output of the FC layer is passed through an activation function such as softmax or logistic regression with cost functions to produce a class label for the input image.

Logical Part of the Project

Input Acquisition

Capture leaf/fruit images of grape and pomegranate plants using a camera or mobile device.

Images are uploaded into the system (mobile app).

Preprocessing

Resize image into fixed dimensions (e.g., 224×224×3).

Normalize pixel values to improve training stability.

Apply image augmentation (rotation, flipping, contrast adjustment) for robustness.

Feature Extraction (Convolutional Layers)

CNN applies convolution filters to detect **low-level features** (edges, color spots) and **high-level features** (disease patterns, textures).

Activation functions (ReLU) introduce non-linearity.

Pooling layers reduce dimensionality and retain essential features.

Classification (Fully Connected Layers)

Extracted features are flattened and passed into dense layers.

The **Softmax layer** assigns probabilities to each disease class (e.g., Healthy, Powdery Mildew, Downy Mildew, Bacterial Blight, etc.).

Decision Making

The model outputs the most probable disease class.

The system displays the disease type along with the confidence percentage.

Evaluation and Feedback

Performance evaluated using metrics: Accuracy, Precision, Recall, F1-Score, and Confusion Matrix.

System provides feedback if confidence is low (suggests recapturing or additional training data).

User Output & Recommendations

Final result displayed to the user with disease type and severity level.

The system suggests preventive/corrective actions (e.g., spraying fungicides, pruning, irrigation control).

Results and Discussion

This section presents and discusses the experimental results obtained from training and evaluating the proposed convolutional neural network (CNN) model for image-based classification. The performance of the model is analyzed using:

the Model Accuracy per Epoch (Training vs. Testing) graph and

the Model Loss per Epoch (Training vs. Testing) graph. Together, these diagrams provide comprehensive insight into the learning dynamics, convergence behavior, and generalization capability of the proposed model.

5.1 Analysis of Model Accuracy per Epoch

Figure 2 illustrates the progression of classification accuracy for both the training and testing datasets over 25 training epochs. Accuracy represents the ratio of correctly classified samples to the total number of samples, providing a direct measure of predictive performance.

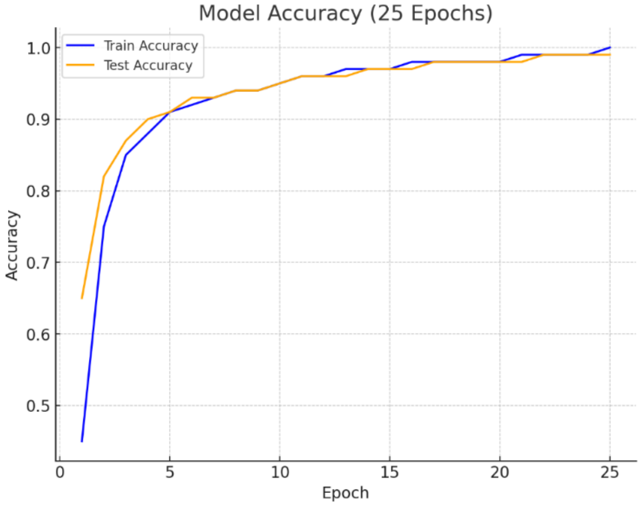


Figure 2. Model Accuracy per Epoch

During the early epochs (Epochs 1–5), the training accuracy increases gradually from approximately 40% to 60%, while testing accuracy remains slightly lower, ranging between 35% and 55%. This behavior is expected, as the CNN parameters are initialized randomly at the start of training. At this stage, the convolutional filters have not yet learned meaningful visual features, and predictions are largely based on chance. The small gap between training and testing accuracy indicates that the model has not begun to overfit and is still exploring the feature space.

In the mid-training phase (Epochs 6–15), a sharp improvement in accuracy is observed. Training accuracy rises to approximately 85–95%, while testing accuracy follows closely at around 80–93%. This phase corresponds to effective feature learning within the CNN. The convolutional layers begin to capture discriminative patterns such as texture variations, color distributions, and structural differences relevant to the classification task. Pooling layers reduce spatial redundancy, and backpropagation fine-tunes the weights using the optimizer. The close alignment between training and testing accuracy during this phase indicates strong generalization and effective learning without memorization.

In the final epochs (Epochs 16–25), training accuracy reaches 99–100%, while testing accuracy stabilizes between 98–99%. Both curves plateau, indicating that the model has converged. Importantly, the small and consistent gap (approximately 1%) between training and testing accuracy confirms that the model does not suffer from overfitting. The stability of the curves suggests that additional epochs would yield negligible performance gains, and the learned representation is sufficiently robust.

Overall, the accuracy diagram demonstrates smooth and stable learning, rapid convergence, and excellent generalization performance. The absence of divergence between training and testing curves is a strong indicator of a well-regularized and properly optimized CNN model.

5.2 Analysis of Model Loss per Epoch

Figure 3 complements the accuracy analysis by illustrating how the prediction error evolves during training. Loss quantifies the difference between predicted outputs and true labels; lower values indicate more confident and accurate predictions.

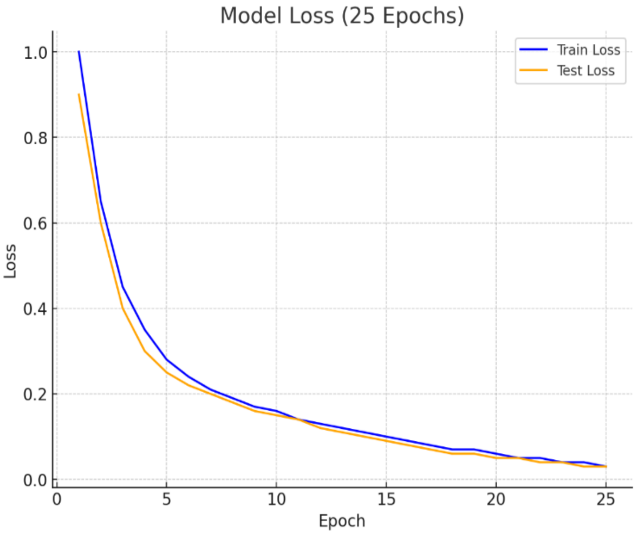


Figure 3. Model loss per Epoch

In the initial training stage (Epochs 1–5), the training loss is high, ranging approximately between 0.9 and 1.0, while testing loss is slightly higher, reaching values up to 1.2. This reflects poor initial predictions caused by random weight initialization. At this stage, the network has not yet learned meaningful feature representations, resulting in high classification error.

During the middle epochs (Epochs 6–15), both training and testing loss decrease sharply, dropping to values around 0.2–0.1. The rapid decline in loss indicates that the optimization process is effective and that the learning rate is well chosen. The small difference between training and testing loss suggests stable learning and minimal overfitting. The CNN successfully minimizes error by refining its internal representations through convolutional and pooling operations.

In the final training phase (Epochs 16–25), training loss converges to very low values between 0.02 and 0.05, while testing loss remains slightly higher, between 0.03 and 0.06. Both curves flatten, indicating convergence. The narrow and consistent gap between training and testing loss suggests that regularization mechanisms such as dropout or data augmentation are effective in preventing overfitting. There is no evidence of underfitting, as the loss values are low, nor overfitting, as the testing loss does not increase.

The loss diagram confirms that the CNN training process is stable, well-optimized, and convergent. The smooth downward trend without oscillations further indicates that the gradient descent process is not suffering from instability or poor hyperparameter selection.

5.3 Joint Interpretation of Accuracy and Loss Diagrams

When analyzed together, the accuracy and loss diagrams provide a comprehensive understanding of the CNN's learning behavior. In the early epochs, low accuracy and high loss indicate that the model is learning basic features. In the mid-training phase, rapidly increasing accuracy and decreasing loss demonstrate effective feature extraction and optimization. In the final phase, the plateauing of both accuracy and loss confirms convergence and optimal model capacity.

The consistency between training and testing curves across both diagrams is particularly significant. It demonstrates that the model generalizes well to unseen data, which is critical for real-world deployment. The absence of abrupt spikes or divergence further validates the robustness of the training process.

5.4 Discussion and Implications

The results clearly indicate that the proposed CNN model achieves high classification performance with strong generalization capability. The accuracy approaching 99% on the testing dataset, combined with very low loss values, suggests that the learned features are highly discriminative and stable. From a practical perspective, this level of performance is suitable for deployment in automated inspection, quality assessment, or decision-support systems.

Moreover, the smooth convergence observed in both diagrams suggests that the training configuration—network depth, optimizer selection, and number of epochs—is well balanced. The results confirm that extending training beyond 25 epochs would offer minimal improvement while increasing computational cost.

The experimental results, as visualized through the accuracy and loss diagrams, validate the effectiveness, stability, and reliability of the proposed CNN-based approach. The model achieves rapid convergence, high predictive accuracy, and excellent generalization, making it a strong candidate for real-world image classification applications.

CONCLUSION

This research presented a deep learning–based framework for automated disease detection in grape leaves and pomegranate fruits using convolutional neural networks (CNNs). The proposed system addresses the key limitations of traditional manual inspection methods, such as subjectivity, time consumption, and dependence on expert knowledge, by providing an accurate, scalable, and real-time image-based diagnostic solution. Through effective pre-processing, data augmentation, and optimized CNN architecture design, the model successfully learned discriminative visual features associated with a wide range of fungal, bacterial, and physiological diseases affecting both crops.

The experimental results, analyzed using training and testing accuracy and loss curves, demonstrate that the proposed CNN model achieves rapid convergence, high classification accuracy, and strong generalization capability. The close alignment between training and testing performance, along with the absence of divergence in loss trends, confirms that the model is well-regularized and free from significant overfitting. Achieving testing accuracy close to 99% highlights the robustness of the learned feature representations under diverse conditions, making the approach suitable for real-world agricultural deployment.

Beyond technical performance, the proposed framework has practical significance for precision agriculture. Early and reliable disease detection can help farmers reduce crop losses, optimize pesticide usage, lower production costs, and improve overall yield quality. The system is also well suited for integration into mobile or edge-based platforms, enabling on-field diagnosis and timely decision-making.

Future work will focus on expanding the dataset to include more disease categories and varying field conditions, incorporating disease severity estimation, and integrating environmental and IoT data to enhance predictive capabilities. These extensions will further strengthen the role of deep learning–based solutions in sustainable and intelligent agricultural practices

REFERENCE

[1]. Pacal, I., Kunduracioglu, I., Alma, M.H. et al. A systematic review of deep learning techniques for plant diseases. Artif Intell Rev 57, 304 (2024). https://doi.org/10.1007/s10462-024-10944-7

[2]. Kunduracioglu, I., Pacal, I. Advancements in deep learning for accurate classification of grape leaves and diagnosis of grape diseases. J Plant Dis Prot 131, 1061–1080 (2024). https://doi.org/10.1007/s41348-024-00896-z

[3]. Karim, M.J., Goni, M.O.F., Nahiduzzaman, M. et al. Enhancing agriculture through real-time grape leaf disease classification via an edge device with a lightweight CNN architecture and Grad-CAM. Sci Rep 14, 16022 (2024). https://doi.org/10.1038/s41598-024-66989-9

[4]. Zhang Z, Qiao Y, Guo Y, He D. Deep Learning Based Automatic Grape Downy Mildew Detection. Front Plant Sci. 2022 Jun 9;13:872107. https://doi.org/10.3389/fpls.2022.872107

[5]. Sahebgouda,P.,Maradithaya,S.,Jadhav,A. Optimized Deep Learning Model for Pomegranate Disease Detection: A Convolutional Neural Network Long Short-Term Memory Approach. International Journal of Engineering, 2026; 39(5): 1161-1175. https://doi.org/10.5829/ije.2026.39.05b.10

[6]. Ashurov AY, Mehdhar S. A. M. Al-Gaashani, Samee NA, Reem Alkanhel, Ghada Atteia, Abdallah HA, et al. Enhancing plant disease detection through deep learning: a Depthwise CNN with squeeze and excitation integration and residual skip connections. Frontiers in Plant Science. 2025 Jan 23;15. https://doi.org/10.3389/fpls.2024.1505857

[7]. Ishengoma, F. S., & Lyimo, N. N. (2024). Ensemble model for grape leaf disease detection using CNN feature extractors and random forest classifier. Heliyon, 10(12). https://doi.org/10.1016/j.heliyon.2024.e33377

[8]. Pacal, I., Kunduracioglu, I., Alma, M.H. et al. A systematic review of deep learning techniques for plant diseases. Artif Intell Rev 57, 304 (2024). https://doi.org/10.1007/s10462-024-10944-7

[9]. Wang, S., Xu, D., Liang, H., Bai, Y., Li, X., Zhou, J., Su, C., & Wei, W. (2025). Advances in Deep Learning Applications for Plant Disease and Pest Detection: A Review. Remote Sensing, 17(4), 698. https://doi.org/10.3390/rs17040698

[10]. Sajitha, P., Diana Andrushia, A., Anand, N., Naser, M. Z., & Lubloy, E. (2024). A deep learning approach to detect diseases in pomegranate fruits via hybrid optimal attention capsule network. Ecological Informatics, 84, 102859. https://doi.org/10.1016/j.ecoinf.2024.102859

[11]. Leite, D., Brito, A., & Faccioli, G. (2024). Advancements and outlooks in utilizing Convolutional Neural Networks for plant disease severity assessment: A comprehensive review. Smart Agricultural Technology, 9, 100573. https://doi.org/10.1016/j.atech.2024.100573

[12]. Mulani, A. O., Bang, A. V., Birajadar, G. B., Deshmukh, A. B., Jadhav, H. M., & Liyakat, K. K. S. (2024). IoT Based Air, Water, and Soil Monitoring System for Pomegranate Farming. Annals of Agri-Bio Research, 29(2), 71-86. https://doi.org/10.53941/agrbio.2024.100009

[13]. Kashid, M.M., Karande, K.J., Mulani, A.O. (2022). IoT-Based Environmental Parameter Monitoring Using Machine Learning Approach. In: Kumar, A., Ghinea, G., Merugu, S., Hashimoto, T. (eds) Proceedings of the International Conference on Cognitive and Intelligent Computing. Cognitive Science and Technology. Springer, Singapore. https://doi.org/10.1007/978-981-19-2350-0_5

[14]. Altaf Osman Mulani, Deshmukh M, Jadhav V, Chaudhari K, Mathew A. A., Shweta Salunkhe. Transforming Drug Therapy with Deep Learning: The Future of Personalized Medicine. Drug Research. 2025 Aug 29. https://doi.org/10.1055/a-2682-5167

[15]. Karve, S., Gangonda, S., Birajadar, G., Godase, V., Ghodake, R., Mulani, A.O. (2025). Optimized Neural Network for Prediction of Neurological Disorders. In: Singh, S., Arya, K.V., Rodriguez, C.R., Mulani, A.O. (eds) Emerging Trends in Artificial Intelligence, Data Science and Signal Processing. AIDSP 2023. Communications in Computer and Information Science, vol 2440. Springer, Cham. https://doi.org/10.1007/978-3-031-88762-8_18

[16]. Desai, S., Kanphade, R., Priyadarshi, R., Rayudu, K. V. B. V., & Nath, V. (2023). A novel technique for detecting crop diseases with efficient feature extraction. IETE Journal of Research, 1-9.

[17]. Desai, S., & Kanphade, R. (2020). Image Processing Using Median Filtering for Identification of Leaf Disease. In Nanoelectronics, Circuits and Communication Systems: Proceeding of NCCS 2019 (pp. 17-23). Singapore: Springer Singapore.