International Journal on Computational Modelling Applications
Vol. 03, Iss. 01, S. No. 01, pp. 01--14, March 2026
ISSN (Online): 3048-8516

Received: 09 Jan 2026, Accepted: 11 April 2026, Published: 20 April 2026
Digital Object Identifier: https://doi.org/10.63503/j.ijcma.2026.218

Research Article

Apple Plant Disease Detection System using Leaf Images

Poonam Chaudhary[1*], Sneha Kandacharam[2] , Akshara Kunda[3], Ayushi Sharma[4], Debopriya Mukherjee[5]

[1, 2, 3, 4, 5] Department of Computer Science Engineering, The Northcap University, Gurugram, Haryana, India

poonamchaudhary@ncuindia.edu

*Corresponding author: Poonam Chaudhary, poonamchaudhary@ncuindia.edu

ABSTRACT

The cultivation of apples is affected by various apple plant diseases. These diseases, if not identified and treated on time, may lead to considerable losses in yield. Early detection is highly essential in order to provide early warnings to farmers and to help in identifying diseases at an early stage so that further action can be done to prevent the spread of disease as these diseases cannot be identified through naked eyes in their early stages. This leads to less wastage of yield. This paper proposes a comparison among the deep learning models such as LetNet, AlexNet, VGG, Resnet, Inception Net, and DensNet, for the efficient classification of leaf diseases of the apple plant. The model is trained on the Plant Village Dataset (Updated) taken from kaggle, which contains both healthy and diseased leaf images of apple plants. In this, images go through various preprocessing techniques like resizing, normalizing, and augmenting images in order to increase the robustness of the model. AlexNet attained the maximum classification accuracy in initial trials but had the largest number of parameters, didn't use modern regularization, and hence got at risk of overfitting at some early stage. The paper improved their performance by using a hybrid architecture which consisted of MobileNetV3 and ResNet50 because MobileNetV3 offered efficient extraction of features with little computational expense. It was further complemented by the depth features offered by ResNet50. The main aim for proceeding with the idea of hybrid architecture was not only to improve generalization but also to prevent overfitting.

The hybrid model is implemented using streamlit. This web interface allows the users to upload images of leaves and get real-time results predicting whether the leaves are affected by a disease or not. The system demonstrates high classification accuracy and effective differentiation among visually similar diseases. However, the model's performance in terms of empirical data analysis is influenced by dataset quality, computational resource demands, and its limited ability to generalize in the presence of sparse data. Despite these challenges, the proposed solution provides a scalable and accessible tool to assist farmers and agricultural experts in early disease detection and management.

Keywords: *Apple Plant, Disease Detection, LetNet, AlexNet, VGG, Resnet, Inception Net, DenseNet*

1. Introduction

Apples are one of the most popular fruits in the world, and apple production has a significant social and economic impact. Apple production provides jobs for millions of people around the world, and it is a major source of income for many families. It also helps to support rural communities, and it can help to improve the quality of life for people in developing countries [1]. As of 2022, global apple production reached approximately 95.84 million metric tons of apples. In India, apple production has been on an upward trajectory. In the fiscal year 2022, the country produced over 2.4 million metric tons of apples. India's domestic apple production in the marketing year 2024/2025 is forecast 6 percent higher at 2.55 million metric tons due to favorable weather conditions in the apple-growing regions. The primary apple-producing regions in India include Jammu and Kashmir, Himachal Pradesh, and Uttarakhand. Jammu and Kashmir contribute significantly to the nation's apple output. *(Food and Agriculture Organisation of the United Nation Survey of 2022)* [28] [29].

The major factors for the low production of apples are ecological factors, poor post-harvest technologies, less thrust on basic research, inadequate supply of quality planting materials to farmers and socio-economic constraints, lack of skills to identify diseases at early stage, etc. Despite their high consumption and medicinal benefits, apple trees are prone to a variety of diseases caused due to insects and microorganisms such as bacteria [1]. This paper focuses on solving the lack of skills to identify diseases at an early stage and the model aims to identify disease in apple plants using machine learning and deep learning algorithms. This model provides an early warning so that the farmers can identify diseases early and take actions to prevent the spread. This ensures less wastage of the yield. This model is also more accurate than manual detection. An automated disease assessment device using black and white images and videos of leaves belonging to potted tomato and blackened fern plants was one of the earliest studies that utilized traditional image processing for plant diseases in 1983. Image processing techniques were also used to quantify streak disease in corn, and it was reported that computer-based methods were more accurate than traditional visual analysis. Some traditional machine learning techniques that were used for plant disease identification include support vector machines (SVM) for tomato disease identification, random forests for tomato disease identification, and K-nearest neighbours (KNN) for soybean disease identification. Machine learning techniques were also used to identify the disease and estimate severity. In addition, SVM, KNN, and Naïve Bayes were used for tomato powdery mildew disease identification [2]. In the past, farmers manually detected crop diseases, relying on experience and observational skills. The method required highly skilled farmers who could recognize symptoms like leaf discoloration, wilting, mold growth, and pest damage. Different ML approaches such as K-means clustering and support vector machine (SVM) have been employed for plant and disease classification and detection. However, due to complex image preprocessing and feature extraction steps, such methods have lower performance and speed in real-time disease detection because of high-dimensionality it gets complex, it treats images as 1-D vectors, ignoring spatial relationships (edges, shapes, textures). Additionally, one of the main drawbacks of traditional ML approaches is that they are not suitable for real-life detection scenarios with non-uniform complex backgrounds [3]. Recent studies on pattern classification reveal a growing interest in deep models and ensemble classifiers. In particular, the combination of deep models for classification tasks has emerged as a highly popular research focus. Deep learning, a subset of machine learning, has become a preferred approach for disease identification due to increased computational power, storage capacities, and availability of large datasets. Common Deep learning techniques include using Convolutional Neural Networks (CNN) for image classification, object detection, and semantic segmentation [2]. Inspired by the breakthroughs of CNNs in object detection, research and applications of CNNs are not rare in crop disease detection currently.

2. Literature Review

Bansal et al. (2021) introduced an ensemble learning model in their paper, Disease detection in apple leaves using deep convolutional neural network, combining DenseNet121, EfficientNetB7, and NoisyStudent (CNN architectures) to classify Healthy, Apple Scab, Apple Cedar Rust, and Multiple Diseases. The model used the approach of transfer learning, in which pre-trained models are there for improving results using less data. DenseNet was selected because of its ability to connect between all the layers, hence reusing the features is easy. EfficientNet was used for its ability to give better accuracy using fewer parameters, and Noisy Student was selected since it uses semi-supervised learning, for improved generalization of the model. Finally, the ensemble model validated a 96.25% accuracy on validation, able to identify multiple diseases with a 90% accuracy. However, this research failed in performing in a real-world environment [1]. Aanis Ahmad et al. (2023), in their paper, A survey on using deep learning techniques for plant disease diagnosis and recommendations for development of appropriate tools, surveyed 70 studies on deep learning for plant disease detection. They evaluated dataset requirements, imaging sensors, preprocessing, and compared traditional ML methods (SVM, KNN, random forests) with DL methods (various CNNs, transfer learning). Although the performance metrics are high, gaps persist in dataset diversity, sensor integration, and tool scalability, emphasizing on the need for more robust and generalized solutions [2].

Arunabha M. Roy and Jayabrata Bhaduri (2021) developed a work, A Deep Learning Enabled Multi-Class Plant Disease Detection, in which they discuss a model that analyzes multiple plant diseases using a computer vision-based deep learning approach. The images of the dataset, apple plant disease, was enhanced using data augmentation and feature extraction pre-processing techniques. The model uses a CNN-based YOLOv4 algorithm that shows satisfactory results with 91.2% mAP and 95.9% F1-score. But the research can be further improved for its real-time implementations [3]. Nachtigall et al. (2016) used CNNs, consisting of architectures like AlexNet, and Multilayer Perceptron (MLP) models to classify various apple leaf diseases (Black Rot, Rust, and Scab). Their study demonstrated the performance of CNNs over traditional machine learning approaches, achieving an accuracy of 97.3% with CNN and 77.3% with MLP. The CNN was used both for feature extraction and classification. However, the dataset was relatively small, collected under controlled conditions with white backgrounds and lacked diversity in lighting, background, and leaf orientation. These limitations reduce real-world applicability. More diverse datasets from natural environments are required for further research to improve model generalization and practical field results [4]. Sharada P. Mohanty et al. (2016) , aim to demonstrate large-scale plant disease detection using deep learning in their work, Using Deep Learning for Image-Based Plant Disease Detection. They train a deep CNN on the PlantVillage dataset (54,306 images) after resizing to 256×256 and image preprocessing with Lab/HSB corrections. The method achieves 99.35% overall accuracy and a mean F1 score of 0.9934. Despite the satisfactory performance results, using controlled images as a dataset limits generalization to field conditions [5].

Yin Min Oo et al. (2018) developed a Plant Leaf Disease Detection and Classification Using Image Processing. In this, diseases like Bacterial Blight, Cercospora Leaf Spot, Powdery Mildew, and Rust were analyzed using a dataset from Myanmar. Preprocessing included noise removal and segmentation, with LBP and GLCM for feature extraction. A supervised SVM model showed high accuracy. The research can be expanded further by using datasets of real-world images and improving real-time detection [6]. Baranwal et al. (2019) developed a LeNet-5-based CNN model for classifying different apple plant leaf diseases such as Black Rot, Rust, Apple Scab, and Healthy Leaves. For increasing the classification accuracy, the model focused on image preprocessing, resizing and normalization. Their study put forward CNN-based feature extraction with the LeNet-5 deep learning architecture that achieved an accuracy of 98.54%. To achieve a high performance accuracy, the training time was estimated to be nearly two hours for 40-50 epochs. It utilized a GTX 1080 Graphics Card with parallel computing and CUDA cores. However, since the dataset was initially collected in controlled laboratory environments, its applicability in real-world conditions was limited. To evaluate robustness in practical scenarios, real-world images dataset could be used rather [7].

Muhammad Hammad Saleem et al. (2019) developed their research, Plant Disease Detection and Classification by Deep Learning. Public datasets like PlantVillage (Kaggle) were used to analyze multiple plant diseases. The images were preprocessed using data augmentation and feature scaling. For extracting features from the images, different CNN models like AlexNet, VGG, and ResNet were used. The deep CNN models achieved an accuracy of ~99%. However, the dataset they used were less diverse and the model needs to improve generalization for being efficient in real-world conditions [8]. Peng Jiang et al. (2019) discussed developing a deep learning model that can detect the apple leaf diseases in real-time. For this, their study mainly focused on five apple leaf diseases, Alternaria, Brown Spot, Mosaic, Grey Spot, Rust. The dataset they were using had a total of 26,377 images. These images were preprocessed by applying data augmentation and annotation. A CNN model, INAR-SSD (SSD with Inception module and Rainbow concatenation) was used, which achieved 78.80% mAP and 23.13 FPS. The study talks about the importance of detecting diseases in real-time which will be efficient for field applications [9]. Chao et al. (2020) in their work, created a model combining DenseNet and Xception-based CNN models. This model helped in identifying diseases like Mosaic, Rust, Grey Spot, Brown Spot, and Alternaria Leaf Spot in apple leaves. DenseNet was used since it has the ability to reuse the features and to connect each layer which enables efficient feature propagation. The Xception model helped in reducing the number of parameters through its depthwise separable convolutions making the model lightweight and efficient. Instead of using the fully connected layers, the model used Global Average Pooling (GAP). It helped the model in improving its feature extraction ability. This combined DCNN (DenseNet + Xception) with an SVM classifier achieved 98.82% accuracy. It showed better performance than some other models like Inception-v3, MobileNet, and VGG-16. However, the model still needs to be trained in diverse backgrounds to make it ideal for practical applications. Also, this study did not discuss the harm that the detected disease could cause [10].

Zhang et al. (2020) worked on a Deep Learning-based Object Detection Improvement for Tomato Disease by using Faster R-CNN with ResNet-101 in his paper to improve feature extraction. ResNet-101 allowed training of a deeper network through skip connections, which improved the learning of complex patterns. Their work compared different models like Faster R-CNN, Faster R-CNN-Mobile, and Faster R-CNN-ResNet101. The models achieved accuracies of 95.83%, 94.67%, and 97.18% respectively. A Region Proposal Network (RPN) was used to refine anchor frames. K-means clustering was applied to determine the required size and scale of anchor boxes by clustering bounding box dimensions. This approach helped to align anchor proposals more closely with real object shapes that further improved boundary localization. Although this study was more inclined towards tomato diseases, this approach could be adapted for apple disease detection [11]. Li et al. (2021) provided a review of deep learning methods for plant disease detection in their work Plant Disease Detection and Classification by Deep Learning-A Review. The objective of the study was to combine CNN architectures, transfer learning, and visualization techniques across various publicly available datasets. They discussed some preprocessing methods like normalization, augmentation, and segmentation, and evaluation metrics like accuracy, precision, recall, F1, mAP, showing satisfactory performance. Despite the result, imbalanced datasets and non-standard evaluation methods are some gaps that have to be addressed further [12]. Muhammad E. H. et al. (2021), in their paper, Automatic and Reliable Leaf Disease Detection Using Deep Learning Techniques, focused on tomato leaf disease detection. They combined U-net segmentation with an EfficientNet-based CNN for classification to achieve their objective. The dataset consisted of 18,161 images, which were preprocessed using segmentation, resizing, and augmentation. The model achieved 98.66% accuracy, 98.5% IoU, 98.73% Dice (segmentation metrics) and classification accuracies up to 99.95%. However, practical implementations due to the lack of real-world validations in the images and cross-crop adaptations [13]. Punam Bedi and Pushkar Gole (2021) developed a hybrid model for detecting diseases in plants. For this hybrid model, they studied various plant diseases whose dataset has not been specified. Their approach included extracting features with the help of CNN and texture-based methods. The hybrid model was created by combining CNN and SVM models. This model achieved 96-98% accuracy. To enhance the computational efficiency and real-time applications, hybrid approaches should be improved [14].

Karpyshev et al. (2021) proposed an Autonomous Mobile robot for Apple Plant Disease Detection based on CNN and a Multi-spectral Vision System. The model was trained on the PlantPathology Apple Dataset (Kaggle) and improved with data augmentation techniques such as cropping, resizing, rotation, flipping, and normalization. The purpose was to detect different diseases such as Apple Rust, Apple Scab, and Moniliasis. The CNN model achieved F1-scores of 0.983 for Healthy Leaves (Precision: 0.992, Recall: 0.975), 0.970 for Rust (Precision: 0.960, Recall: 0.980), and 0.972 for Scab (Precision: 0.972, Recall: 0.972). However, its dependency on RGB data limits early disease detection capability [15]. Bi et al. (2022) in their research, Multi-model LSTM-based convolutional neural networks for detection of apple diseases and pests, introduced a MobileNetV2-based deep learning model for detecting Alternaria Leaf Blotch and Apple Rust in apple trees. The approach aimed to lower the computational cost and ease of use on mobile devices. MobileNetV2 is a lightweight CNN which was used for both feature extraction and classification. It fine-tuned the model with fewer data, achieving accuracy closer to more complex models like ResNet152 and InceptionV3 but with lower computational cost. The method was designed to be stable and capable of identifying similar apple leaf diseases. It provided accurate detection in less time, making it efficient and precise. A larger and more diverse set of images could be used for training to improve the performance of model [16]. Turkoglu et al. (2022) in their research paper MobileNet based apple leaf diseases identification developed a Multi-Model LSTM-based CNN model to identify apple diseases and pests in Turkey. In their hybrid model, they used various pre-trained CNN architectures of AlexNet, GoogleNet, DenseNet201 combined with Long

Short-Team Memory (LSTM) layers which capture temporal dependencies and improved the sequential feature learning. This ensemble model achieved an accuracy of 99.2%, outperforming other models like VGG16, ResNet18, ResNet50, ResNet101, and InceptionResNetV2. But it takes a lot of time to train high-dimensional feature vectors and find optimal parameters for LSTM architecture. The model struggles in the real-world environment and is not suitable for large datasets even after achieving high accuracy [17].

Zhu et al. (2022) created Apple-Net, an improved version of YOLOv5-based object detection model for detecting various apple leaves spots and diseases such as Alternaria Blotch, Brown Spot, Gray Spot, Mosaic and Rust. In their Apple-Net model, they used an improved YOLOv5 version as it is better for real time detection and gives better efficiency resulting in more accurate and fast object detection. They used Feature Enhancement Module (FEM) to identify small and overlapping objects as it captures more contextual information and for better localization of objects, they used Coordinate Attention (CA). For enhancing image quality in conditions like rain or poor lighting, the model used Generative Adversarial Network (GAN). The model achieved better results 95.9% mean Average Precision (mAP@0.5) and 93.1% precision than conventional object detection models. This study does not focus on detailed disease analysis as it is mainly enhanced object detection. It can be improved by focusing on detecting and analysing specific disease characteristics. [18]. Asif Iqbal Khan et al. (2022) in their research paper Deep Diagnosis: A Real-Time Apple Leaf Disease Detection System Based on Deep Learning, developed a two-stage system which focused on detecting apple leaf diseases such as scab, alternaria, mosaic. They used transfer learning and data augmentation in a lightweight CNN model and used object detection to detect spots on apple leaves. They achieve 88% classification accuracy and 42% mAP using 9000 expert-annotated RGB images with augmentation. However, the method could not detect very small disease spots, highlighting a need for enhanced field robustness [19]. Anju Yadav et al. (2022), in her paper, AFD-Net: Apple Foliar Disease Multi Classification using Deep learning on Plant Pathology dataset, targeted apple foliar diseases such as scab, rust, and powdery mildew. They used a modified EfficientNet-based CNN including transfer learning and augmentation. The approach achieves 98.7% and 92.6% accuracy for datasets Plant Pathology 2020 and Plant Pathology 2021 respectively. Further validation in real-field conditions is required to address overfitting and generalizability issues [20].

C. Jackulin and S. Murugavalli (2022) reviewed different literatures on plant disease detection, and compared traditional ML (e.g., SVM, KNN with handcrafted features) with DL (CNN-based) methods in their work, A comprehensive review on detection of plant disease using machine learning and deep learning approaches. The objective was to check the performance across different studies using various datasets and preprocessing methods (normalization, augmentation, segmentation). Aggregated evaluation metrics (accuracy, F1 score, precision, recall, mAP) showed that the DL methods performed better. However, challenges still exist with standardized datasets, early detection, and model interpretability [21]. Laura Falaschetti et al. (2022) created a CNN-based Image Detector for Plant Leaf Diseases Classification. PlantVillage and ESCA datasets were used to analyze various plant leaf diseases. To train the model, they use various preprocessing methods which include data augmentation and tensor decomposition, they also choose CNN for feature extraction. They achieved 98.10% accuracy using a supervised CNN model but for real-time application suggest optimizing CNNs [22]. Vibhor Kumar Vishnoi et al. (2022) made a model for identifying apple leaf diseases such as Apple Cedar Rust, Apple Scab and Black Rot. They used the PlantVillage dataset which is available in Kaggle. Preprocessing methods such as data augmentation and hyperparameter tuning, along with CNNs were used for efficient extraction of features. The supervised CNN model achieved 98% accuracy. For future work it is suggested to test the model beyond the PlantVillage dataset and in a real-world environment [23].

Bulent Tugrul et al. (2022) used Convolutional Neural Networks in Detection of Plant Leaf Diseases: A Review. Various plant diseases were studied across multiple datasets. CNN-based models like VGG, ResNet, and Inception were used for feature extraction. Supervised learning was applied, achieving over 95% accuracy. The study mentions inaccuracy in computation and hence the need for lightweight CNN models for practical applications [24]. Md. Manowarul Islam et al. (2023), discuss developing a real-time web application for crop disease prediction in their work, DeepCrop: Deep learning-based crop disease prediction with web application. Their objective was achieved by evaluating several CNNs on the PlantVillage dataset (~10,000 images) with standard preprocessing (resizing, normalization, augmentation) and transfer learning via ResNet-50, which yields 98.98% accuracy. However, the system requires training on more diverse datasets to achieve accuracy and robustness in varied field conditions [25].

Acharya, V., & Ravi, V. (2024) developed an improved version of Capsule Neural Network (CapsNet) for detecting Apple Rust, Apple Scab, and multiple other apple plant diseases. CapsNet is better due to its ability to capture spatial relationships and provide more accurate information about the object's position, angle, and scale through capsules that output vectors indicating these properties. This helped the model to better understand whether the object (disease) is in the right place or not, hence improving classification accuracy. They got an accuracy of 91.37% which was 3.67% more than previous methods on the same dataset as they improve feature learning by using image enhancement techniques. One of the main limitations is in its generalizability and applicability in the real-world agricultural field as this model is not tested in the real-world environment. This model can be improved by focusing on using it in a diverse and real-world environment [26]. Munaf Mudheher Khalid and Oguz Karan (2024) in their research paper Deep Learning for Plant Disease Detection, they compared a custom CNN with MobileNet to detect plant diseases early, incorporating GradCAM for explainability. They use a public dataset with standard image preprocessing techniques like resizing, normalization, augmentation. They achieved 89% accuracy with 96% precision, recall, and F1 for the custom CNN, and 96% accuracy (with slightly lower metrics) for MobileNet. To address the problem of diverse disease types effectively, there is a need for improving generalization and transfer learning. [27].

The objective of this research is to compare the performance of different deep learning models for disease classification on apple plant leaves and then identify a model, most efficient for early and accurate detection. In this paper, we propose the implementation of a hybrid deep learning model using MobileNetV3 and ResNet50 for classifying the diseases on apple leaves to achieve improved generalization and reduced overfitting.

3. Dataset and Methodology

Disease-infected plants usually show obvious marks on leaves, stems, flowers, or fruits. Generally, each disease or pest condition presents a unique visible pattern that can be used to uniquely diagnose abnormalities [12]. There are several diseases which attack apples, the major one being cedar apple rust (Gymnosporangium juniperivir ginianae), scab (Venturia inaequalis) and black rot (Botryosphaeria obtusa) [1].

Black rot leaves *Cedar apple rust* *Scab*

Figure 1: Random sample of diseased apple leaves

3.1 Dataset Description:

The Plant Village Dataset from Kaggle was used. This dataset consists of 9714 high-quality images of diseased and healthy plant leaves of the apple plant. Figure 1 shows some samples of diseased apple leaves. It has 3 data splits (train, test, and validation), with consistent categories across all splits. This dataset is ideal for machine learning researchers and practitioners working on plant disease detection and classification, as well as for agricultural experts seeking to improve plant health and crop yields. From this dataset, this research included specifically Apple dataset which consists of three types of apple diseases (Apple Scab, Black Rot, Cedar Apple Rust) and healthy apple leave:

Train: (107.3MB)

- Apple Scab: 2016 images (27MB)
- Black Rot: 1987 images (35.6MB)
- Cedar Apple Rust: 1760 images (17MB)
- Healthy: 2008 images (27.7MB)

Test: (23.9MB)

- Apple Scab: 453 images (6.06MB)
- Black Rot: 447 images (7.96MB)
- Cedar Apple Rust: 396 images (3.79MB)
- Healthy: 451 images (6.13MB)

Validation: (2.61MB)

- Apple Scab: 51 images (651kB)
- Black Rot: 50 images (884kB)
- Cedar Apple Rust: 44 images (423kB)
- Healthy: 51 images (722kB)

Table 1 provides information about a curated image dataset of healthy and diseased apple leaves, organized into separate train/validation/test directories.

Table 1: Dataset description

Class Name	**Train**	**Val**	**Test**	**Total**
Healthy	2008	51	451	2510
Apple Scab	2016	51	453	2520
Cedar Apple Rust	1760	44	396	2200
Black Rot	1987	50	447	2484
Total	7771	196	1747	9714

3.2 Preprocessing:

All images were resized to 224 × 224 pixels to match the input size expected by MobileNetV3 and ResNet50 architecture. During preprocessing, pixel values were normalized to the [0, 1] range by scaling each channel through division by 255. To improve model robustness and reduce overfitting, real time data augmentation using ImageDataGenerator was employed on the training set with the following transformations: random rotations within ±45°, translations up to 30% of image dimensions, distortions up to 30%, random zooming within the range [0.7, 1.3], random horizontal and vertical flips, brightness values randomly adjusted in the range [0.5, 1.5], pixels exposed by transformations were filled using nearest-neighbor interpolation. Additionally, the model used in-model augmentation using Keras preprocessing layers, such as random flip (horizontal), random rotation (20% of 360°), random zoom (20%). Moreover, the two-fold augmentation approach, which relies on external augmentation at the time of data loading as well as internal augmentation in the model, has been an integral part in improving the generalization capability of the models. Both the validation set and the test set were normalized between the range [0-1], and no augmentation was applied to preserve original distribution.

3.3 Proposed Model Architecture

The proposed model's architecture can be described by a pipeline that represents the workflow of an image classification task in deep learning. This hybrid model is created by combining the MobileNetV3 and ResNet50 models. After preprocessing, the images are first passed through a MobileNetV3 feature extractor. This extractor is important since it is responsible for learning important visual patterns. It encodes basic visual information, from low-level edges, corners, simple shapes to mid-level such as textures, leaf veins and spots into a compact 7x7x960 feature map. Then, the features are passed through a bottleneck layer which is implemented using 1x1 convolution, batch normalization, and residual addition. This layer is responsible for enhancing the important extracted features and optimizing the feature maps while maintaining computational efficiency.

Next, the features are passed through the ResNet50 feature extractor, which uses the previous features (output from the MobileNetV3) to understand the patterns already detected and then this extractor learns the semantic information which will help it differentiate between the apple leaf diseases. The extractor is able to learn and differentiate between patterns by ResNet's skip connections. After processing, the image information is reduced to 7x7x2048 feature map. Here, an increase in the number of channels (2048) means that the model has more complex features. In order to enhance the focus of the model on the more important features further, the Squeeze-and-Excitation block performs global pooling and attention to adaptively re-weight the channels of the features with respect to their relevance. The final output is fed into a classifier consisting of a dense layer (1024), dropout for avoiding overfitting, and softmax activation that generates class probabilities. The two phases of training are used to maximize the effectiveness of performance; phase 1 has only the top layers trained while keeping the backbones frozen; it allows stable learning up to 30 epochs. Phase 2 fine-tunes the top layers of the backbone supported with callbacks-early stopping and learning rate reduction along with model checkpoints for better convergence and preventing overfitting. Regularization techniques such as dropout, L2 weight regularization, and label smoothing are used to stabilize the model and improve generalization. For training the model, the Adam optimizer is used along with an initial learning rate of 1e-4 and 1e-5 during fine-tuning for adaptive and smooth learning. The test set metrics used to evaluate the trained model are accuracy, confusion matrix, and ROC curve to comprehensively assess classification performance.

4. Experimental Setup, Results and Discussion

The model combines two architectures, ResNet50 and MobileNetV3. MobileNet, a CNN architecture, emerged from Google in 2017, targeting efficient image processing on mobile and embedded platforms. By leveraging depthwise separable convolutions, the computational expense gets significantly reduced [27]. Hyperparameter tuning is an important aspect for finding the optimal values of external parameters. The hyperparameters can be determined before the models are trained. They control the performance, the training speed of the models, and the way they perform when generalizing the models. This paper incorporates manual hyperparameter tuning through thoughtful selection and adjustment of various training parameters. A custom learning rate scheduler adjusts the learning rate across different training phases: (i) Warm-up phase: 1e-4 for initial 5 epochs, (ii) High learning rate phase: 1e-4 for the next 15 epochs, (iii) Reduced learning rate phase: 5e-5 for the subsequent 20 epochs, (iv) Fine-tuning phase: 1e-5 for the remaining epochs. Additionally, ReduceLROnPlateau callback reduces the learning rate by a factor of 0.5 if the validation loss plateaus, with a minimum learning rate threshold of 1e-7. It utilizes the batch normalization in the batch size of 32, standard power-of-two that balances speed and generalization. Also, up to 100 epochs but stop after 10 with no validation loss improvement using early stopping setting in the code. Hyperparameter-like regularization with dropout (0.5 → 0.3) and L2 (0.001) to prevent overfitting and label smoothing (0.1 → 0.05) to reduce overconfidence was experimented. Utilizing freeze both backbones for 30 epochs which lead to new head warms up and unfreeze only top layers afterward to refine high-level features. Figure 3 provides the best trial hyperparameter values.

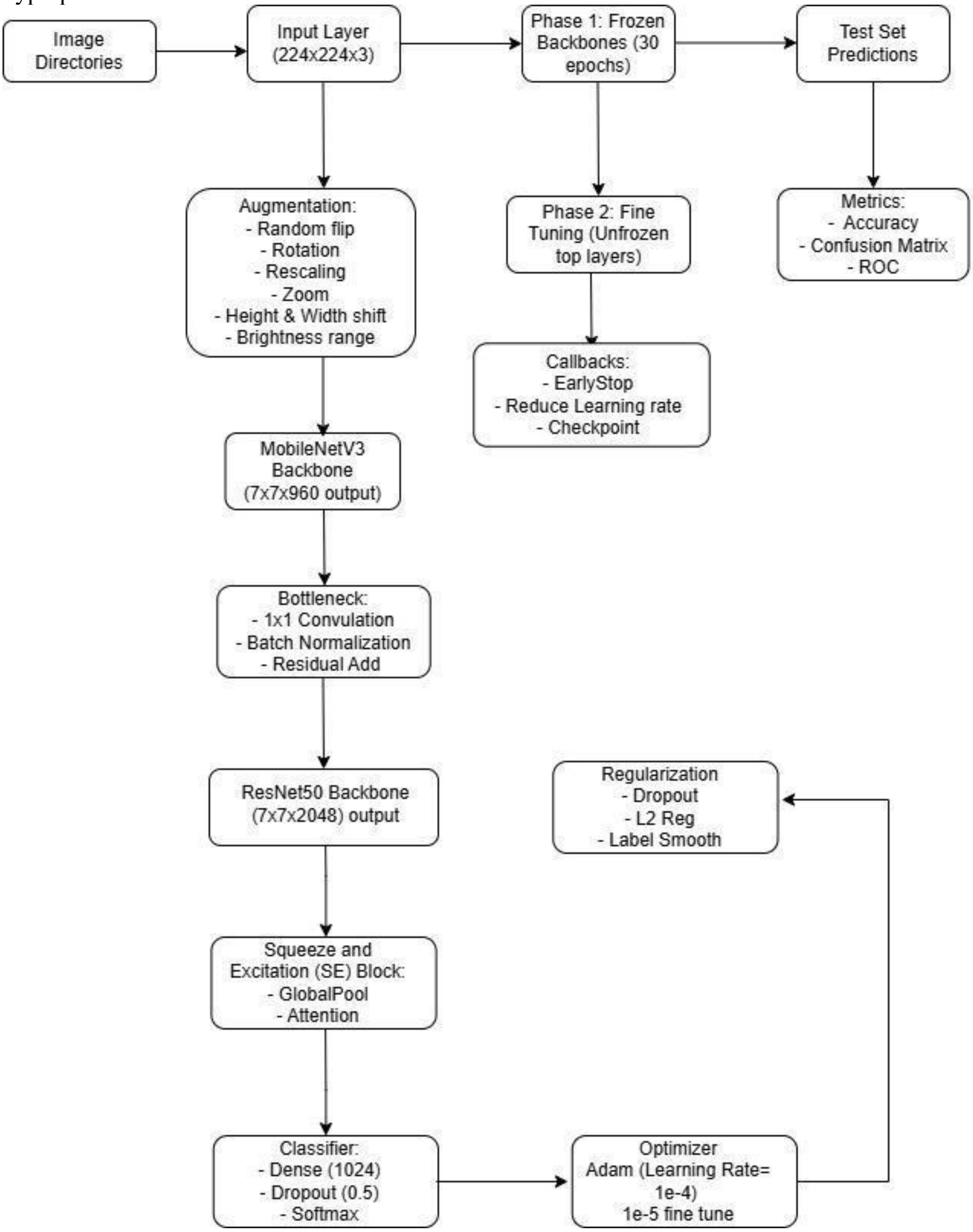


Figure 2: Proposed Model Architecture

```
=== Best Trial ===
Hyperparameters:
learning_rate: 0.0001
dropout_rate: 0.3
l2_reg: 0.0001
batch_size: 32
label_smoothing: 0.1
```

Figure 3 : Hyperparameters used

5. Training

ResNet 50 and MobileNetV3 are used with Transfer learning. Transfer learning (TL) is an ML strategy where knowledge from one task is leveraged to improve performance on a related, subsequent task. This technique adopts a model pre-trained on a particular problem to tackle a different but associated challenge. The core principle of TL is the portability of knowledge. Insights and feature patterns extracted from one context can provide a head start when approaching a new problem, often reducing the computational cost and time to train [27]. Layers are kept "Frozen", which means its weights are initially not updated during training (common strategy in TL). Later, during fine-tuning, the top layers (or even more) are unfrozen and trained further on our dataset. This enables the adaptation of the high-level features of the model in a way that is more apt for the task. The training phase for this hybrid system is such that it aims for maximum performance without resulting in overfitting. It is a two-part process:

Phase 1: Feature Extraction (Frozen Backbones):

Both MobileNetV3 and ResNet50 backbones are frozen in order not to break the pretrained knowledge. Only the classifier head is trained over a few epochs, such as 20 or 30, in such a way that the model learns from the specific dataset with a minimum chance of disrupting learned representations. Data augmentation is applied during training to increase generalization. Adam optimizer along with with a learning rate around 1e-4. and categorical cross-entropy with optional label smoothing.

Phase 2: Fine-Tuning (Unfreeze Top Layers):

The top layers of ResNet50 and MobileNetV3 are unfrozen for further training to refine feature representations specific to the task. A lower learning rate (e.g., 1e-5) is used to prevent large weight updates. Callbacks like Early Stopping, ReduceLROnPlateau, and ModelCheckpoint are applied to monitor validation performance and avoid overfitting. Regularization techniques (e.g., Dropout, L2 weight decay) help maintain model robustness. Table 2 provides the comparison of different architectures in terms of overall accuracy. Whereas figure 4 shows the confusion matrix for better result analysis.

Table 1: Comparison of Architectures

Model	DL Architecture	Accuracy (%)
LeNet	Classic CNN: 2 Convolutional layers + 2 Fully Connected (FC) layers	97.76
AlexNet	5 Convolutional layers + 3 Fully Connected layers, ReLU, Dropout, MaxPooling	99.49
ResNet	Residual Learning using skip connections	98.47
VGG	Deep CNN with 13 Convolutional layers + 3 FC layers	96.94

Inception Net	Parallel convolutional paths with varying kernel sizes	50.51
DenseNet	Dense connectivity: each layer receives input from all previous layers	99.08
MobileNet V3 + ResNet50	Hybrid: Combines MobileNetV3's lightweight, fast Feature extraction with ResNet50's deep contextual learning	99.77

AlexNet (99.49 %) has achieved the best overall accuracy. Its combination of relatively large early kernels (11×11), aggressive dropout (50 %), and max-pooling proved especially effective at capturing both global shape and local texture cues in leaf images. Whereas **DenseNet-121** with the dense connectivity encourages feature reuse, which helps with gradient flow and parameter efficiency results in 99.08 % . However, it requires more GPU memory and longer training time than AlexNet. Residual skip-connections of **ResNet** help mitigate vanishing gradients in deeper networks resulting in 98.47 %, but the deeper variants did not outperform AlexNet on this moderately sized dataset. LeNet still performs respectably with **LeNet** 97.76% despite its simplicity. VGG-16/19 underperforms relative to AlexNet, likely due to its large number of parameters and lack of aggressive regularization. The parallel-path design of **Inception-v1/v3** may have over-complicated feature extraction for this relatively small and homogeneous dataset, leading to overfitting or optimization difficulties. **MobileNetV3** offers improved efficiency and accuracy via optimized blocks, Hard Swish activation, and NAS-driven design, making it superior for mobile and edge device image tasks with limited resources. NAS-driven design refers to using Neural Architecture Search (NAS) to automatically discover or optimize the structure (architecture) of a neural network.

After implementation of different CNN architectures on the dataset, AlexNet emerged as the best model for apple leaf disease classification but due to overfitting in AlexNet, a combined model has been created using **ResNet50 and MobileNetV3** for improved accuracy and efficiency. Figure 5 gives a sample Predicted Class for an Apple Scab leaf image while testing a new sample on the trained model.

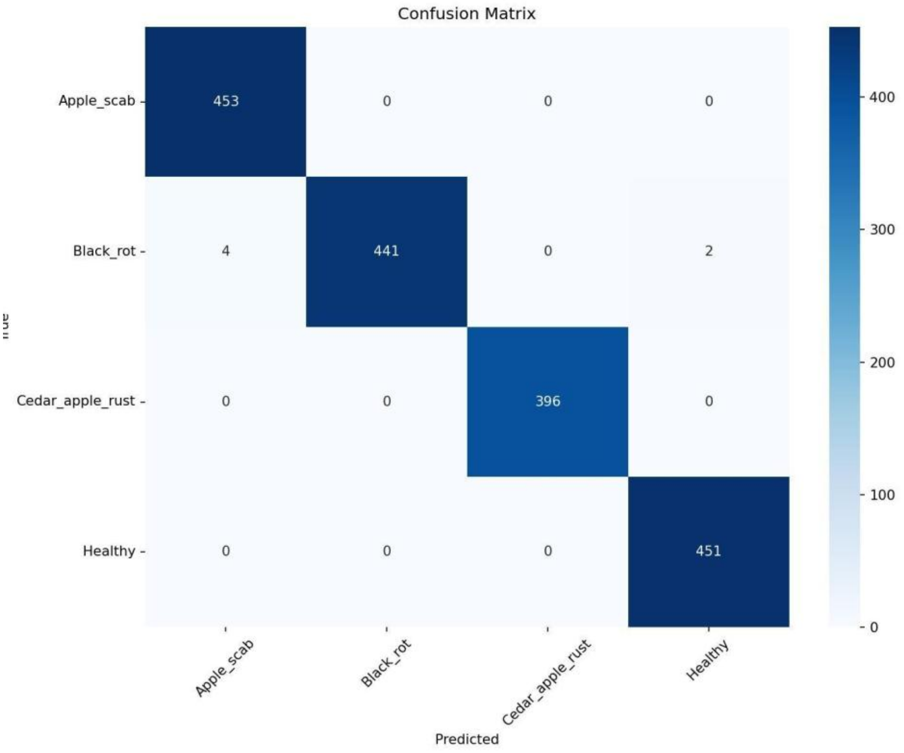
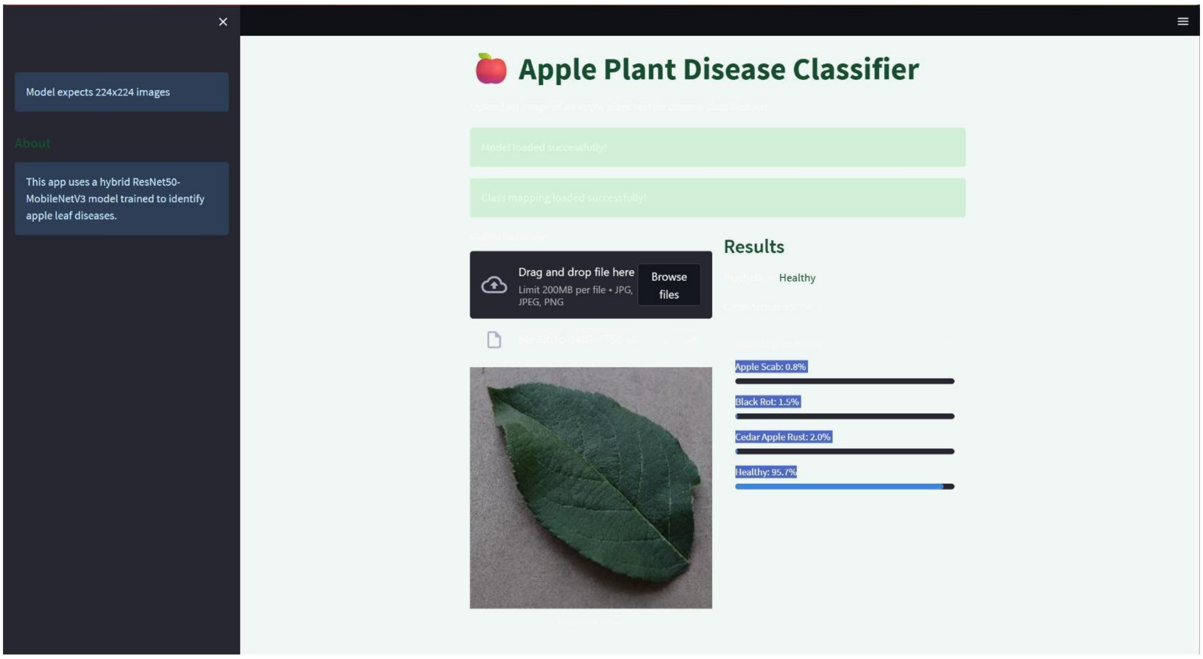


Figure 4: Confusion matrix

To make the trained model accessible to end users, the inference pipeline was compiled into a web application using Streamlit. Streamlit's simple Python API makes it easy to build an interactive UI with file-upload widgets and real-time display of prediction probabilities. The application accepts an uploaded apple leaf image, applies the required resizing and normalization, and then feeds the preprocessed image into the saved model. At the backend, the model is loaded once at startup for efficiency, and each new image is processed on the server GPU (or CPU). This requires only a single Python script and can be deployed on any cloud platform that supports Streamlit, making our disease-detection model easy to share with farmers and plant-health experts without using any additional front-end development. Figure 5 and 6 gives the glimpse of real time image processing with proposed model deployment.



Model expects 224x224 images

About

This app uses a hybrid ResNet50-MobileNetV3 model trained to identify apple leaf diseases.

Apple Plant Disease Classifier

Model loaded successfully!

Class mapping loaded successfully!

Results

Healthy

Apple Scab: 0.8%

Black Rot: 1.5%

Cedar Apple Rust: 2.0%

Healthy: 95.7%

Figure 5: Web Interface using Streamlit

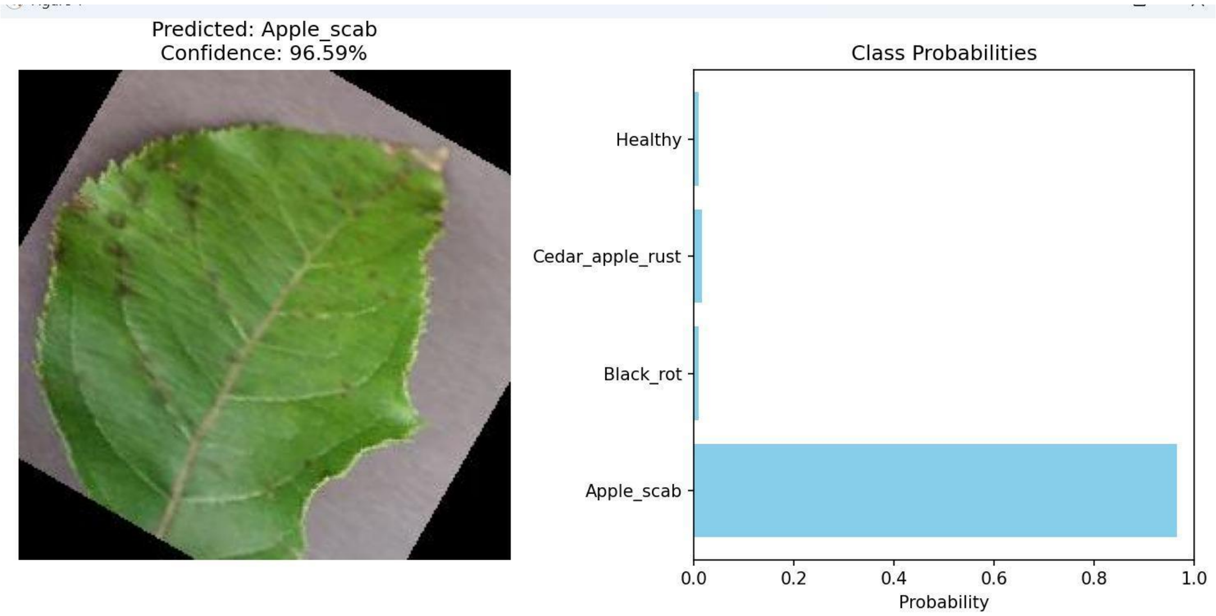


Figure 6: Predicted Class for an Apple Scab leaf image

6. Conclusion

The selection of a deep learning-based model for apple plant disease classification, in our case, is a combined model of ResNet50 and MobileNetV3 architecture. This selection is justified by the model's proven efficiency and capability in handling complex image classification tasks. The strength of this combined CNN approach is that it utilizes the strong feature extraction and feature learning capabilities of ResNet through residual connections and the efficiency that MobileNet performs the same task using fewer computations and less memory, making the overall model faster and more efficient.

For improving the quality of data input and helping the model to learn better, image preprocessing is applied that leads to enhancement in the quality of the images fed to the model. Object detection is used further for identifying the spots on the leaves so that the model can classify the disease. For making the model even better in terms of accuracy, hyperparameter optimization is performed. This helped the model improve accuracy by considering the best possible parameters. All these approaches together helped in developing a deployable model that has the ability to support the detection of diseases at an earlier stage, thereby facilitating the scalability of agricultural management.

Conflict of Interest

The authors declare that there is no potential conflict of interest.

References

[1] P. Bansal, R. Kumar, and S. Kumar, "Disease detection in apple leaves using deep convolutional neural network," *Agriculture*, vol. 11, no. 7, p. 617, 2021.

[2] Ahmad, D. Saraswat, and A. El Gamal, "A survey on using deep learning techniques for plant disease diagnosis and recommendations for development of appropriate tools," *Smart Agricultural Technology*, vol. 3, p. 100083, 2023.

[3] M. Roy and J. Bhaduri, "A deep learning enabled multi-class plant disease detection model based on computer vision," *AI*, vol. 2, no. 3, pp. 413–428, 2021.

[4] L. G. Nachtigall, R. M. Araujo, and G. R. Nachtigall, "Classification of apple tree disorders using convolutional neural networks," in *Proc. IEEE 28th Int. Conf. Tools with Artificial Intelligence (ICTAI)*, 2016.

[5] S. P. Mohanty, D. P. Hughes, and M. Salathé, "Using deep learning for image-based plant disease detection," *Frontiers in Plant Science*, vol. 7, p. 1419, 2016.

[6] Y. M. Oo and N. C. Htun, "Plant leaf disease detection and classification using image processing," *International Journal of Research and Engineering*, vol. 5, no. 9, pp. 516–523, 2018.

[7] S. Baranwal, S. Khandelwal, and A. Arora, "Deep learning convolutional neural network for apple leaves disease detection," in *Proc. Int. Conf. Sustainable Computing in Science, Technology and Management (SUSCOM)*, 2019.

[8] M. H. Saleem, J. Potgieter, and K. M. Arif, "Plant disease detection and classification by deep learning," *Plants*, vol. 8, no. 11, p. 468, 2019.

[9] P. Jiang, Y. Chen, B. Liu, D. He, and C. Liang, "Real-time detection of apple leaf diseases using deep learning approach based on improved convolutional neural networks," *IEEE Access*, vol. 7, pp. 59069–59080, 2019.

[10] X. Chao *et al.*, "Identification of apple tree leaf diseases based on deep learning models," *Symmetry*, vol. 12, no. 7, p. 1065, 2020.

[11] Y. Zhang, C. Song, and D. Zhang, “Deep learning-based object detection improvement for tomato disease,” Mar. 2020. [Online]. Available: https://ieeexplore.ieee.org/document/9044330

[12] L. Li, S. Zhang, and B. Wang, “Plant disease detection and classification by deep learning—a review,” *IEEE Access*, vol. 9, pp. 56683–56698, 2021.

[13] M. E. H. Chowdhury *et al.*, “Automatic and reliable leaf disease detection using deep learning techniques,” *AgriEngineering*, vol. 3, no. 2, pp. 294–312, 2021.

[14] P. Bedi and P. Gole, “Plant disease detection using a hybrid model based on convolutional autoencoder and convolutional neural network,” *Artificial Intelligence in Agriculture*, vol. 5, pp. 90–101, 2021.

[15] P. Karpyshev *et al.*, “Autonomous mobile robot for apple plant disease detection based on CNN and multi-spectral vision system,” in *Proc. IEEE/SICE Int. Symp. System Integration (SII)*, 2021, pp. 157–162.

[16] M. Turkoglu, D. Hanbay, and A. Sengur, “Multi-model LSTM-based convolutional neural networks for detection of apple diseases and pests,” *Journal of Ambient Intelligence and Humanized Computing*, vol. 13, no. 7, pp. 3335–3345, 2022.

[17] Bi *et al.*, “MobileNet based apple leaf diseases identification,” *Mobile Networks and Applications*, pp. 1–9, 2022.

[18] R. Zhu *et al.*, “Apple-Net: A model based on improved YOLOv5 to detect the apple leaf diseases,” *Plants*, vol. 12, no. 1, p. 169, 2022.

[19] I. Khan *et al.*, “Deep diagnosis: A real-time apple leaf disease detection system based on deep learning,” *Computers and Electronics in Agriculture*, vol. 198, p. 107093, 2022.

[20] Yadav *et al.*, “AFD-Net: Apple foliar disease multi classification using deep learning on plant pathology dataset,” *Plant and Soil*, vol. 477, no. 1, pp. 595–611, 2022.

[21] Jackulin and S. J. M. S. Murugavalli, “A comprehensive review on detection of plant disease using machine learning and deep learning approaches,” *Measurement: Sensors*, vol. 24, p. 100441, 2022.

[22] L. Falaschetti *et al.*, “A CNN-based image detector for plant leaf diseases classification,” *HardwareX*, vol. 12, p. e00363, 2022.

[23] V. K. Vishnoi *et al.*, “Detection of apple plant diseases using leaf images through convolutional neural network,” *IEEE Access*, vol. 11, pp. 6594–6609, 2022.

[24] Tugrul, E. El Fatimi, and R. Eryigit, “Convolutional neural networks in detection of plant leaf diseases: A review,” *Agriculture*, vol. 12, no. 8, p. 1192, 2022.

[25] M. M. Islam *et al.*, “DeepCrop: Deep learning-based crop disease prediction with web application,” *Journal of Agriculture and Food Research*, vol. 14, p. 100764, 2023.

[26] V. Acharya and V. Ravi, “Apple foliar leaf disease detection through improved capsule neural network architecture,” *Multimedia Tools and Applications*, vol. 83, no. 16, pp. 48585–48605, 2024.

[27] M. M. Khalid and O. Karan, “Deep learning for plant disease detection,” *International Journal of Mathematics, Statistics, and Computer Science*, vol. 2, pp. 75–84, 2024.

[28] “Agricultural Research Data Book 2024,” 2024. [Online]. Available: https://iasri.icar.gov.in/agridata/23data/Previous%20Year%20Books/Databook2024.pdf

[29] R. Mani, “Fresh Deciduous Fruit Annual,” No. IN2024-0052, 2024. [Online]. Available: https://apps.fas.usda.gov/newgainapi/api/Report/DownloadReportByFileName?fileName=Fresh%20Deciduous%20Fruit%20Annual_New%20Delhi_India_IN2024-0052.pdf