Spectrum of Engineering Sciences
ISSN (e) 3007-3138 **(p)** 3007-312X
Volume 4, Issue 4, 2026

ENHANCING POTATO LEAF DISEASE DETECTION USING HYBRID CNN ARCHITECTURES: ACOMPARATIVE STUDY OF VGG16, RESNET50, AND MOBILENETV2

Yi Weiguo[*1], **Hurair Ali**[2]

[*1,2]*School of Railway Intelligent Engineering, Dalian Jiaotong University, Dalian, 116028, Liaoning Province, China*

[*1]jiekexun98@163.com, [2]hurairali321@gmail.com

DOI: https://doi.org/10.5281/zenodo.19883623

Keywords
Potato Leaf Disease Detection, Image Classification, Deep Learning, Transfer Learning

Article History
Received: 02 March 2026
Accepted: 11 April 2026
Published: 29 April 2026

Corresponding Author: *
Yi Weiguo

Abstract

An investigation studies deep learning methods that use image classification to automate detecting potato leaf diseases. The research combines a created Convolutional Neural Network along with pre-trained VGG16 MobileNetV2 and ResNet50 models to increase performance level on a restricted agricultural dataset through transfer learning approaches for early disease identification in improving crop sustainability and yield.

The research methodology includes data preparation before training models to evaluate their performance by standardized measurement of accuracy alongside precision, recall, F1-score and AUC. The pre-trained ResNet50 achieves the highest accuracy and best robustness along with pre-trained models surpassing the custom CNN based on results produced by performance tests. MobileNetV2 demonstrates the best results for achieving computational efficiency at optimal performance levels. The research has shown that deep learning technology offers real-time smartphone-based agricultural disease diagnosis capabilities to farmers through its viable solutions. The research proposes innovative next steps which include augmenting the dataset to improve its diversity as well as enhancing lightweight model performance and mobile application deployment for practical agricultural use.

INTRODUCTION

Global economy is highly reliant on agriculture and for the developing countries in particular, it is a major contributor to the livelihood of a large section of the population. Plant diseases represent a serious threat to crop production and decrease yield quality as well as quantity. The potato (Solanum tuberosum), a staple food in many areas of the world, is one such crop that is susceptible to many diseases [3]. Late Blight and Early blight are the most common and damaging diseases of potato plants and occur under humid conditions badly damaging the plants.Detection of disease in plants using traditional methods is labor intensive, time consuming, and often needs expert knowledge. The increasing demand of automated systems for the detection of plant diseases contributes towards it. More recently, they have seen promising developments in deep learning-based approaches for automating disease detection from image data. CNN with pre-trained models VGG16, MobileNetV2 and ResNet50 have advanced to become superior techniques for plant disease detection tasks in image classification domains. Image-based disease classification in plants achieves high accuracy through the extraction of advanced features by the models. These models achieve disease and health state classification of potato leaves through extensive training with large image datasets which enables automatic disease detection technology development.

1.2 Problem Statement

This research focuses on developing an automated deep learning-based system for the detection of potato diseases (Late Blight and Early Blight) in images of potato leaves. Deep neural network power will be used by the system and the performance of some models such as VGG16, MobileNetV2, ResNet50 and CNN are going to be evaluated based on their ability to accurately identify these diseases [12]. The dataset used for training and testing the models consists of images that are classified into three categories: Potato_healthy, Potato_Early_blight, and Potato_Late_blight. Deep learning models have been successful in similar tasks but further investigation of their efficacy against plant disease detection, and in particular potato disease, is required. This research will address the following key questions:

1. With VGG16, MobileNetV2, and ResNet50, how popular CNN architectures perform in potato leaf disease detection?
2. Do these pre trained models achieve better accuracy and efficiency compared with a custom CNN model?
3. In this regard, I would like to analyse the evaluation metrics of these models, including accuracy, precision, recall, F1 score, and AUC, in classifying potato diseases.

1.3 Primary Objectives

The objectives of this research are:

To Develop and Train Deep Learning Models: Detect potato leaf disease using pre trained CNN architectures (VGG16, MobileNetV2, ResNet50) and a custom CNN architecture.

To Evaluate Model Performance: Models evaluation on the standard evaluation metrics such as precision, F1 score, recall, support, accuracy, confusion matrix, ROC curve, AUC, to check whether these models can be used for the classifying potato diseases.

To Compare and Analyze the Results: Models performance were compared, in terms of their classification results and efficiency, and the most suitable model for real world deployment in potato disease detection is identified.

To Implement on Multiple Datasets: Evaluating generalization ability of models on two different datasets of potato diseases and robustness on two different datasets.

1.4 Impact of the Study

This research seeks to further automate agricultural practices through plant disease detection. Deep learning models to detect disease can hugely reduce the time and effort needed for field inspections thus increasing the fast disease management decisions and decrease economic cost on potato farming. Yet, the results from this study are extendable to other crops, allowing the methodology to be applied in the context of different agricultural diseases.

Literature Review

In recent years, there has been considerable interest in detecting and classifying plant diseases using machine learning methods because such techniques are important for efficient and automatic agronomy. CNN, VGG16, ResNet and MobileNetV2 have achieved great success in image-based plant disease classification tasks. It reviews the existing literature of plant disease detection (but especially potato diseases) and discuss various machine learning and deep learning models relevant to plant and crop disease detection.

2.1 Plant Disease Detection Using Machine Learning

Over the past decade, machine learning (ML) and deep learning (DL) have brought tremendous development in plant disease detection. In plant pathology, the use of machine learning applications found its early beginnings in the use of these traditional algorithms including Support Vector Machines (SVM), Random Forests (RF), and k-Nearest Neighbors (k-NN). These methods usually relied on extracting handcrafted features, like texture, color and shape of plant leaves, to do the classification. However, traditional time and frequency domain-based methods work on fixed dataset based on assumptions, yet with high variability in plant images due to environmental factors, lighting conditions and developmental stages of diseases, those methods show lower detection accuracy in real world agricultural settings. The introduction of CNN will be a breakthrough in this field. Unlike tradition methods CNN are capable of learning hierarchical feature representations directly from raw image data without any manual feature extraction activity. This ability to learn from and adapt to complex patterns of large image datasets has also dramatically improved the accuracy of plant disease detection, including the classification of different diseases in the crops.

For example, Atila et al. in 2021 showed that deep learning models, in particular CNN, were effective for the plant disease classification from leaf images. CNN are being used with great success in plant disease classification within agriculture tasks, achieving greater than 99% accuracy rate on a given dataset with images from different crop species including potato leaves. This success has allowed many similar techniques and deep learning methodologies to become adopted in many other areas of agricultural applications.

2.2. Potato Disease Classification Using Deep Learning Models

The Potato Distemper, also known as late blight of potato and early blight of potato, is the most devastating disease of potatoes in the world causing a big threat to food security and economic losses. The early detection and classification of these diseases is important for proper management and timely intervention. Deep learning (DL) has recently seen recent advancements that enables the tool for automation of detection and classification of potato diseases, using convolutional neural networks (CNN) and transfer learning techniques.

In a study by Alzakari et al. in 2024 a CNN-based model was developed to classify potato leaf images into three categories: Late Blight, Early Blight, and healthy. By training the model on a labelled potato leaf image dataset, the researchers were able to achieve high accuracy for classification of the images thus proving the effectiveness of CNN for potato disease detection. Over the years, the study highlighted CNN ability to extract robust features which led to differentiation among visually similar disease symptoms.

Similarly, Khasay el al. in 2019 used the transfer learning techniques with pre-trained models e.g. VGG16 and ResNet50 to detect the potato diseases. However, fine-tuned these models—pre trained on the large-scale ImageNet dataset—on a smaller, labeled potato disease dataset and ended up achieving pretty remarkable results. In that context, transfer learning was found to be beneficial by requiring lower quantities of labelled data whilst still achieving high classification accuracy. In one more interesting study, Arya and Singh in 2019 investigated using AlexNet and GoogLeNet for potato disease as well as other plant diseases classification. They showed that deep learning models trained on sufficiently diverse datasets do generalize well across different crop species and diseases, reaching the overall accuracy of 99.35% for disease classification.

2.3 The Role of Deep Learning Models for Plant Classification Disease

Deep learning systems have become the leading choice in plant disease detection because they deliver better results than older machine learning models. The main structure of today's successful networks comes from the CNN which show excellent results in agricultural image classifications. Large scale models are used extensively to detect and classify diseases in plants using high quality image inputs.

2.4. VGG16 for Plant Disease Classification

VGG16 was used as a deep CNN architecture, comprising a total of 16 layers, comprising convolutional, pooling, and fully connected layers. The model can learn hierarchical features effectively due to its uniform consecutive structure of small (3x3) convolutional filters. Shovon et al. carried out a pioneering study using VGG16 on a large dataset of 14 crop species and 26 plant diseases plant leaf images. The model obtains accuracy above 98% from the test dataset indicating robustness and generalizability for plant disease classification tasks.

Yasin and Fatima (2023) applied VGG16 as well as other deep learning models to detect plant diseases by using a dataset containing 87,848 labelled leaf images. Across various crop types, VGG16 received strong provisional results, averaging a classification accuracy greater than 97%, and a performance more than conventional machine learning methods. Although effective, VGG16 cost can be high from both the use of fully connected layers and from the network's depth, and so is less suitable for deployment on resource constrained environments.

2.5 Custom CNN Models in Agriculture

Besides relying on already known architectures like VGG16, MobileNetV2 or ResNet, researchers have been designing more and more tailored CNN models for particular agricultural tasks, e.g. plant disease detection [65]. They usually include domain specific feature such as leaf shape, color patterns, and texture, which make their classification more accurate for plant health assessment.

For example, Khan et al. (2020) proposed a custom CNN model to classify potato leaf diseases using potato plant image dataset itself is relatively small. The researchers mitigated the issue of limited labeled data by applying advanced data augmentation techniques, including flipping, rotation, and contrast making so as to artificially expand the dataset. This approach resulted in better generalization of the model and better performance on unseen data.Jonnalagadda et al. custom CNN architecture had fewer layers than large pre-trained models such as ResNet or VGG16, which made for computationally efficient and workable in low resource environments. The simplicity of the model did not preclude it from achieving classification accuracies equivalent to those of state-of-the-art pre trained networks, showing that domain specific customization can be achieved in such applications. Custom CNN models are still gaining popularities since they are able to strike an efficient balance between computation costs and accuracy and are tailored for real world applications such as precision farming or on field disease monitoring systems. Simulators also allow for flexibility of incorporating certain features or constraints, e.g. the ability to work on edge devices or flexibility in adapting to specific regional crop varieties.

Methodology

In the methodology, a detailed description of the approach used for classification of potato leaf diseases through deep models is explained. It only contains methods for data acquisition and data cleaning, selection of models, design of models, training of models, and testing of the pre-existing as well as developed models of CNN. This methodology gives a clear and detailed account of the measures followed and applied in the experiment to evaluate and compare different kinds of models in plant disease classification.

3.1 Data Collection

This research used images from PlantVillage repository (Kaggle) from two different datasets to perform its experiments. PlantVillage offers one of the most common open databases for agricultural disease detection studies. This collection offers a wide selection of labeled images showing plant leaf disease variations plus normal leaf images. This research project used only the potato crop photos from the available image collection. It studied three types of potato leaf images in my experiments: entirely healthy specimens and those with early and late blight damage. The objective was to create a problem that classified plant diseases based on particular crops.

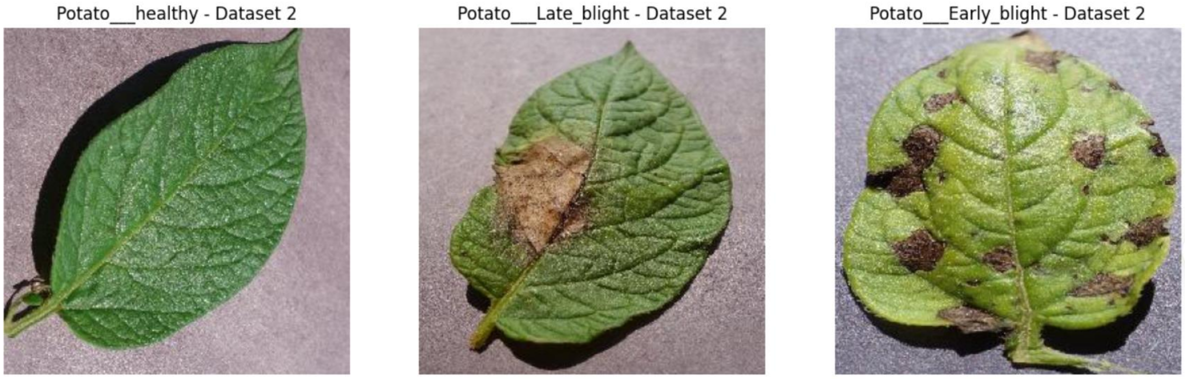

Figure 1: Class Distribution

Table 1: Total Images in Dataset 1

S. No	Category	Number of Images
1	*Potato___healthy*	*500*
2	*Potato___Late_blight*	*500*
3	*Potato___Early_blight*	*500*

Table 2: Total Images in Dataset 2

S. No	Category	Number of Images
1	*Potato___healthy*	*152*
2	*Potato___Late_blight*	*1000*
3	*Potato___Early_blight*	*1000*

It captured about 2,150 images by making sure all three disease types received similar numbers of samples [72]. The right balance of records prevents training biases and lets the model discover important traits for all disease groups. characteristics of each class. The validation set made up 15% of data used to decide the best model parameters while watching training progress to avoid the model fitting too closely to training data. 15% of the images became our test The model needed testing on new data so the dataset split into three sections with 70% training samples 15% validation samples and 15% separate test samples. The training group received 70% of the images to learn the distinct visual set to check how well-trained models could predict results. The organization of the data groups helped us obtain valid results that revealed how the model would work under actual conditions.

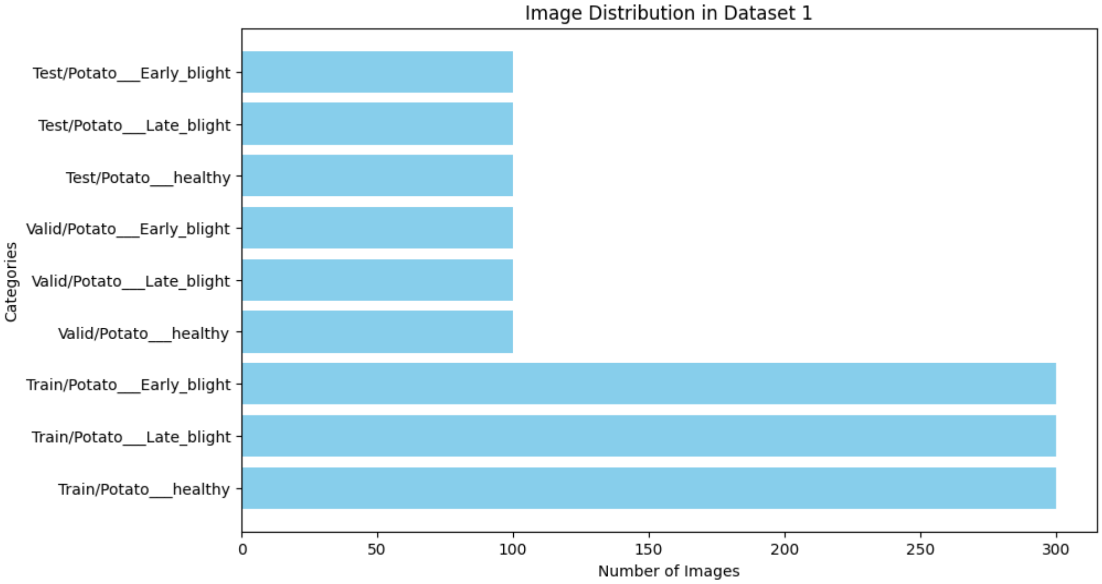

Figure 2: Image Distribution from Dataset 1

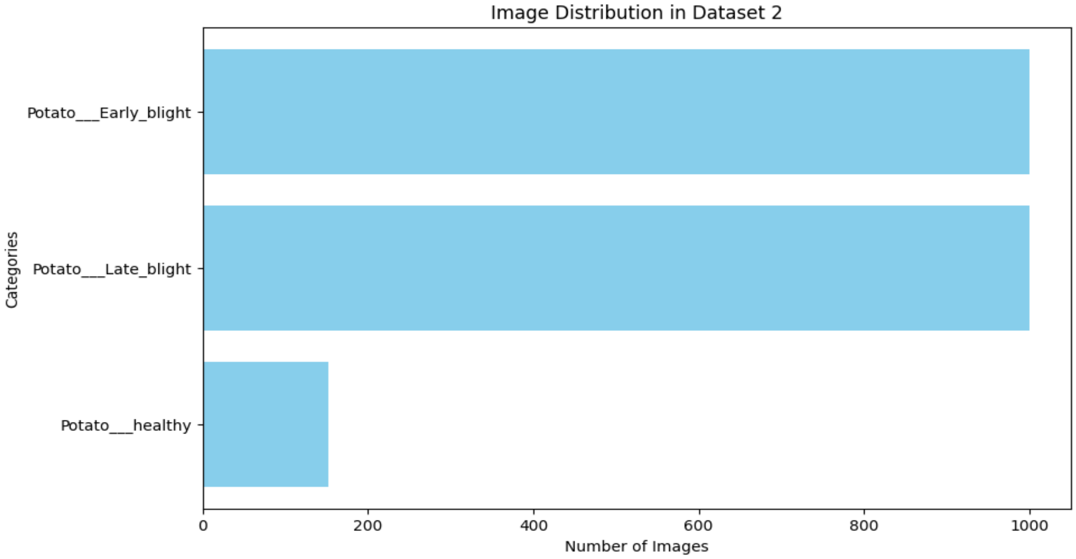

Figure 3: Image Distribution from Dataset 2

3.2 Data Processing

As a preliminary step prior to inputting the images into the deep learning models, there were a number of pre-processing techniques exercised to ensure that the data was clean and standardized, so as well as improve the learning of the models, and reduce the likelihood of overfitting [74]. The datasets and images that were obtained from PlantVillage originated in various sizes and formats, meaning that they had to be preprocessed to fit the input of the selected CNN. Hence, all images were resized to have the size of 224pixels * 224 pixels, which is a standard input format acceptable for CNN systems like VGG16, MobileNetv2, and ResNet50. This preprocessing made the resize work homogenize the dimensions of an image across the dataset so that the models trained prior could easily pull-out features from the similar sizes.

After that, the pixel intensities which were in a scale of [0, 255], were scaled down to [0,1]. This was done to scale the pixel values in order to normalize them since the pixel values ranged from 0 to 255. Normalization is a common data preparation technique that is applied to machine learning, particularly deep learning models, with a view of standardizing the innovational scaling of the data entries, which makes training a faster process. To reduce generalization of the model and as a result of the small size of the dataset augmentation approaches were utilized. These augmentations contained horizontal flip, rotation, zooming, width and height shift, and slight twist. In this way, building environment conditions were replicated when introducing such transformations; thus, it became more resistant to variations in the leaf position, orientation, and lighting. This step was especially useful in minimizing the overfitting of a model, in which the model was able to predict the training data nearly perfectly. The whole data preparation step was performed in Python programming language to provide an interface for the TensorFlow or Keras deep learning library. These offered high-level API for; Loading, pre-processing and augmentation of the images as well as batch-wise feed to the model during training stages. It was facilitating the preprocessing of data inside the model training process, so that all the preparations and the data delivering were consecutive and timely, saving memory and computation time.

3.3 Model Selection

To assess the predictive models in diagnosing potato leaf diseases, this study used four CNN based model; VGG16, MobileNetV2, ResNet50 and custom CNN. These models span in terms of the scale and design principles from purely practical and efficient ones up to the more advanced yet robust network residual connections and specific application-oriented designs.

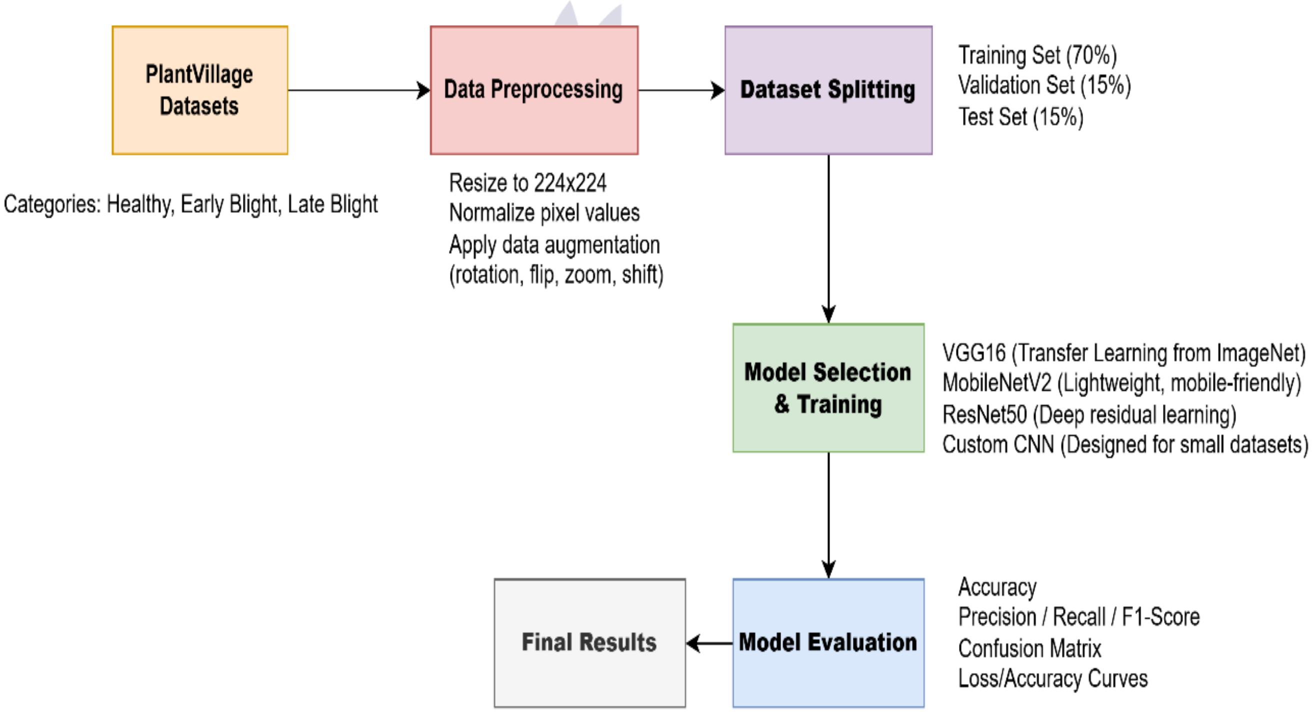

Figure 4: Flow Diagram – Methodology

3.3.1. VGG16 Model

The deep neural network became well-known because it offered simple design and exceptional ImageNet benchmark performance [15]. The weight layers comprise 16 elements that incorporate 13 convolutional components with

ReLU activation following each layer while also containing 3 fully connected layers [86]. The VGG16 model design relies on tiny 3×3 filters that operate across images to extract minute spatial features from the input data. Each block within the design uses two or three convolutional layers before a max-pooling layer reduces spatial dimensions to keep essential features. The flattened output from these five blocks continues to three dense layers until a softmax activation completes the classification stage. The research adopted the VGG16 model through Keras API with TensorFlow backend which received transfer learning modifications to classify potato leaf diseases. The researcher initialized the neural network with weights from the ImageNet dataset because it enables the model to recognize basic visual features. This research used a new layer collection instead of the 1,000 ImageNet-targeted original classifier to categorize Healthy, Early Blight and Late Blight samples. All convolutional blocks near the end as well as the new dense layers became trainable but earlier frozen layers maintained their learned generic low-level features. The combined strategy cut down project completion time while providing greater recognition accuracy because the available information set remained limited.

The performance needed enhancement and overfitting needed mitigation through dropout regularization between fully connected layers. Early stopping and learning rate reduction callbacks served as integrated components during training because they ensured the best possible convergence.

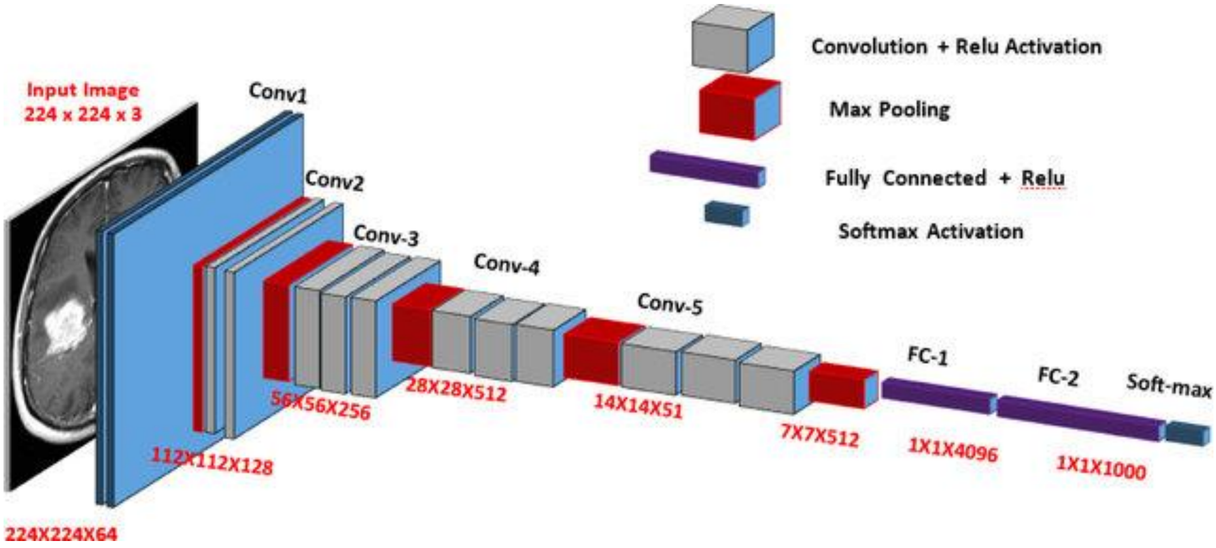

Figure 5: VGG16 Model Architecture

The research adopted VGG16 because it showed good accuracy while being easy to use and displaying practical success in agricultural computer vision applications. The built-in structure of this network design turned it into the perfect solution for extracting distinctive features from potato leaf picture data.

3.3.2. MobileNetV2 Model

MobileNetV2 is a mobile-first convolutional neural network model introduced by Google AI to be optimized for reducing the computational requirements to run efficiently on comparably limited platforms including mobile and edge devices. It is an improved version of MobileNetV1 and includes some primary architectural changes such as the depth wise separable convolution and inverted residual connections with linear bottlenecks which made the MobileNetV2 slimmed and faster with slightly lower accuracy as compared with its predecessor. The key component of MobileNetV2 is the depth wise separation convolution which divides convolution into two parts; depth wise convolution, which uses one filter per channel in the input, and pointwise convolution, which uses a 1*1 convolution to combine results of the depth wise layer. This indeed minimizes the number of computations as compared to the conventional convolutions.

Another primary advantage of MobileNetV2 is that it uses an inverted residual structure. In this setup, the input is first transformed to a higher dimension, then convolved using depth wise convolution operation and is in the end transformed back to a lower dimension by a linear layer. This form of designing a model ensures that important information is saved while at the same time the model is not complex. Residual connections (Skip connections) between the input as well as the output of some blocks help in passing the gradients during the backpropagation process thereby making it easier to train deeper architectures.

The MobileNetV2 was adopted as this network architecture has a good balance of high accuracy and relatively low computational requirements that are crucial for on-field potato disease detection, for example, on smartphones or drones used by farmers [100]. The model was trained from the scratch using pre-trained weights of ImageNet and is trained on potato leaf dataset containing Healthy, Early Blight, Late Blight classes. In the fine-tuning case, the old final classification layer of the network was replaced by a new one – dense layer, which is appropriate for the three classes. The convolutional layers in the first two stages were set to be non-trainable to allow the network to retain the basic ability to recognize images, while the last layers were adjusted to learn how to identify the specifics of potato diseases on the potato leaves. This makes it possible for the training to be of great help in generalization even if the data set that was used in the experiment were relatively small.

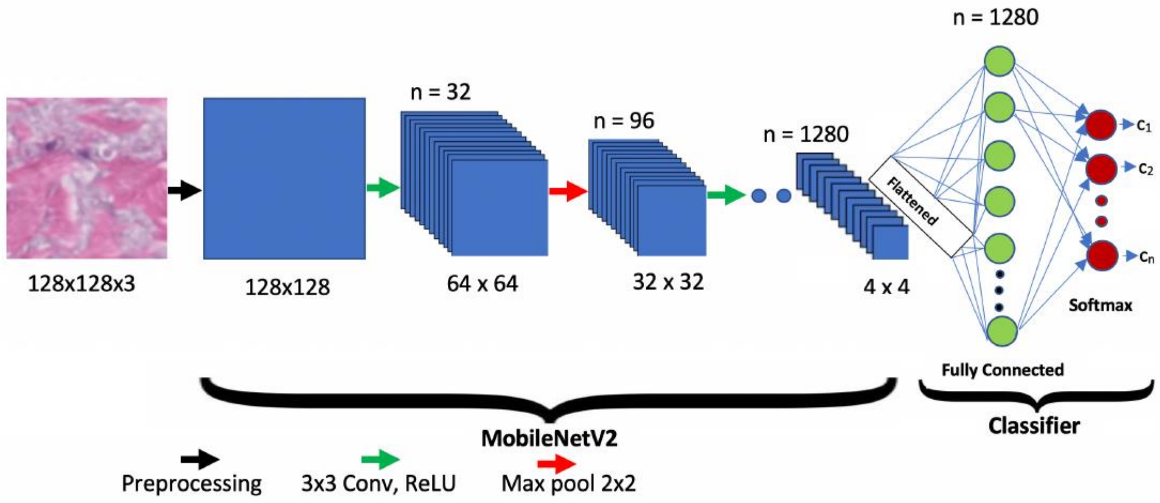

Figure 6: MobileNetV2 Model Architecture

To ensure the model is more efficient and also prevent overfitting, data extrapolation methods were used to increase the available amount of data and dropout was applied to layers of fully connected layers. Batch normalization has been applied in the network to reduce fluctuation and improve the speed of training. MobileNetV2 was found to be an ideal choice for this research in this case not only because of the accuracy it has but most importantly because of its applicability especially on handheld devices used in agriculture. This creates the framework of large-scale and affordable methods of disease diagnosis that can be deployed to remote areas without access to reliable infrastructure.

3.3.3. ResNet50 Model

This deep convolutional neural network architecture by Koonce et al. operates as ResNet50 which targets the degradation problem in deep network training. Network depth expansion normally leads to performance deterioration because vanishing gradients and complex optimization tasks arise. Residual learning through skip connections helps ResNet50 solve the degradation problem by allowing it to learn residual mappings (input-output block differences) instead of complete transformations.

Through residual connections the network finds it simpler to calculate the input-to-output difference compared to mastering the complete transformation itself. The skip connections support efficient gradient propagation during backpropagation thus ensuring constant stability levels especially during ResNet50 training which operates with 50 layers. The ResNet50 model demonstrates both efficiency and stability while processing extensive datasets while maintaining its performance quality unchanged.

The ability of ResNet50 to extract sophisticated hierarchical components from input data makes it exceptionally suitable for plant disease detection work. Through its substance the model successfully detects intricate high-level patterns that distinguish moody potato leaf attributes from normal ones. It becomes essential for shallower networks to detect features that ResNet50 performs well since these features enable accurate disease diagnosis of Early Blight and Late Blight which show subtle symptoms. The research employed ResNet50 because it demonstrates superior ability to acquire complex hierarchical features that recognize both wide-scale and local patterns in image inputs. The ImageNet-pretrained weights of the model received additional training to meet the requirements of potato disease category classification. The design process for disease classification incorporated a new dense layer specific to health and the two forms of blight infection: Early Blight and Late Blight. The model's lower components remained static during initial training to protect general image characteristics obtained from ImageNet yet the upper parts learned specific leaf identification patterns. Although deep in structure ResNet50 incorporates two essential elements; batch normalization alongside dropout that stabilize training and prevent overfitting because they matter greatly for small dataset applications. Training performance received enhancement through integrated features that maintained model robustness across different lighting conditions and leaf textures which commonly exist in agricultural image classification systems

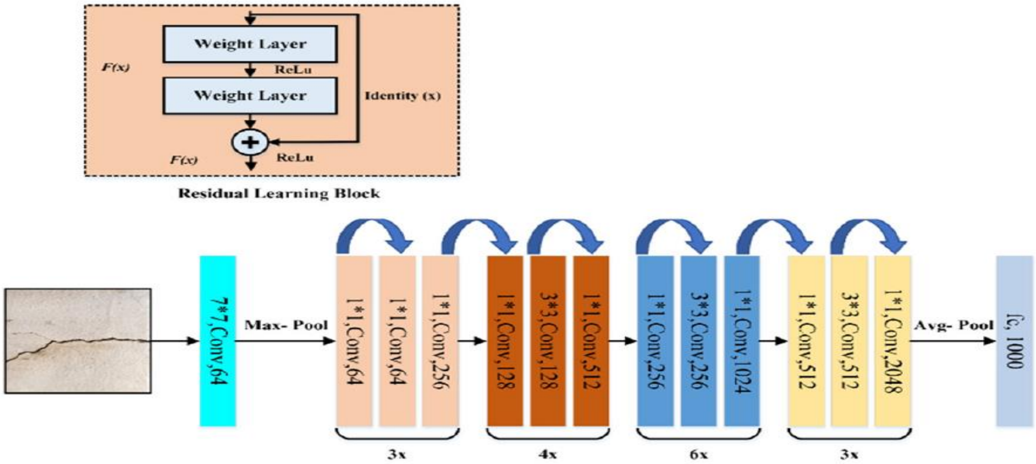

Figure 7: ResNet 50 Model Architecture

The deep structure along with stability features of ResNet50 effectively supports complex disease detection needs like those present in this study because precise and specific classification proves essential. The model effectively distinguished potato disease stages together with types which led to better performance in detecting symptoms and outdone simpler models such as VGG16 and MobileNetV2. The architecture of ResNet50 combined with its residual connections establishes it as a top candidate for high-precision disease detection workloads while demonstrating practical potential for agricultural disease detection in the field.

3.4 Model Evaluation Metrics

Multiple evaluation criteria were required to measure the effectiveness of deep learning models used for potato leaf disease detection. These statistical measurements show both how well the models function and how well they identify different types of diseases. Our tests used these metrics to assess model performance.

3.4.1 Accuracy

The basic measure of accuracy reports how many test set instances a model correctly identifiers among all identified instances. It is defined as:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

where:

True Positives refer to correct classifications between each unique class in TP. The accuracy measure calculates the percentage of instances which both tests and verifies as non-disease classes (TN) [132]. FP describes the number of instances that belong to other classes yet are incorrectly categorized as belonging to a previous group. The term FN defines instances which classifiers placed into incorrect classes. Accuracy functions well as an overall metric but it lacks specificity for imbalanced datasets together with critical tasks that require class-weighted evaluation hence precision along with recall and F1-score measurement becomes necessary.

3.4.2 Precision

Precision identifies the right prediction results among this group [133].

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

3.4.3 Recall

The Sensitivity test (also True Positive Rate or Sensitivity) finds the number of actual positives that the system correctly identifies [134].

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

3.4.4 F1-Score

F1-score finds its value as the balanced average between precision and recall metrics.

$$F1 = 2 \times \frac{Precision + Recall}{Precision \times Recall}$$

When working with class distributions where amounts vary, the F1-Score provides a useful metric because it evaluates precision and recall alongside each other to generate a better assessment metric than accuracy provides independently.

3.4.5 Confusion Matrix

A vital tool named confusion matrix helps users understand how well their model identifies different classes. The matrix shows the precise breakdown between correct and incorrect predictions for the three disease groups (Healthy, Early Blight, Late Blight).

The confusion matrix shows exact error types to identify which diseases the model mistakes each other and enables better performance through adjustments of model parameters or increased training data.

3.4.6 Training and Validation Loss Curves

Through these curves you can observe if the model trains effectively and find potential overfitting problems. Training loss shows how well the model matches training data while validation loss demonstrates if the model can apply its learning to new data. The model effectively remembers training data instead of mastering useful patterns when it achieves good results on training data but performs poorly on validation data.

Regular changes in training results show overfitting when validation results start to rise. To stop the model from overfitting it used training techniques including early stopping, dropout, and data enhancement during the training stage. The performance of our model was evaluated through loss functions at every epoch to show its improvement and make sure it escaped from overfitting.

3.4.7 Evaluation Setup

The TensorFlow and Keras frameworks served to build and train all models which employed high-speed GPU systems for efficient computational operations. GPU acceleration boosted training performance together with model evaluation speed because these technical features suited the deep architecture of ResNet50 and intricate nature of model training complexity. The training proceeded with early stopping activation that used validation loss to halt the process before overfitting occurred during a preset number of epochs. Model hyperparameter optimization was achieved through grid search or random search procedures for identifying the ideal learning rate and batch size parameters as well as other key parameters which delivered the best model outcomes.

3.4 Python Language

High level, interpreted programming language, python is very simple, readable and flexible language. Due to its universal applicability, Flutter is used widely for web development, data science, artificial intelligence, machine learning and deep learning, for a variety of apps. Python is incredibly versatile and hassle free to use, which is why it is a good choice for researchers and developers; especially when you have to work with AI and ML algorithms as it facilitates rapid development and customization of complex algorithms.

It is one of the biggest advantages of Python that it has the vast ecosystem of libraries and frameworks [142]. Powerful tools to build, train and deployment of deep learning models are available in the form of libraries such as TensorFlow, Keras, PyTorch and Scikit-learn [142]. Besides, Python has a clean syntax and a dynamic typing system that makes it friendly to begin and still has the advanced features for seasoned developers. There are many things that make Python standout among other popular programming languages when compare them:

Table 3: Comparison of Programing Languages

Features	Python	C++	Java	R
Syntax	Simple and readable	Complex and less readable	Moderate complexity	Simple but specialized for stats
Performance	Slower due to interpretation	High performance (compiled)	Moderate (JVM overhead)	Moderate to high
Ease of Learning	Very easy	Hard to learn for beginners	Moderate	Easy for statisticians

Libraries	Extensive (TensorFlow, PyTorch)	Extensive (Boost, OpenCV)	Extensive (Spring, Hibernate)	Excellent for data analysis
Community Support	Large, with active contributions	Large but focused on system-level programming	Large, especially for enterprise solutions	Specialized in data science
Use Cases	AI, ML, Web development	System programming, Games	Enterprise apps, Android dev	Data analysis, Statistical computing

3.4 Google Colab IDE

Google Colab functions as a browser-based integrated development environment (IDE) which runs Python code free in cloud environments for its users. Data science along with machine learning projects choose Google Colab because the platform gives users easy Google Drive access combined with TensorFlow and Keras libraries and universal free computational resources including GPUs and TPUs.Google Colab provides users with a convenient working platform. Users can execute Python code directly in the cloud through a system that does not need local installation of libraries or configuration setups [145]. Notebooks remain highly useful for team-based work since members can share them effortlessly. The integration between Colab and Google Drive ensures that users can maintain automatic cloud storage of their work.

Table 4: Comparison of different IDEs

Feature	**Google Colab**	**Jupyter Notebook**	**PyCharm**	**Visual Studio Code (VS Code)**
Environment	Cloud-based (browser)	Local installation	Local installation	Local installation
Resource Access	Free GPUs and TPUs	Depends on local resources	Limited free options	Depends on local resources
Ease of Use	Very user-friendly	User-friendly	Beginner-friendly with setup	Moderate learning curve
Performance	Limited by cloud resources	Depends on local hardware	Can be very powerful on high-end machines	Moderate performance on local resources
Collaboration	Easy to share (Google Drive)	Requires manual sharing	Collaboration via version control	Collaboration via version control
Support for Machine Learning	Excellent (with GPU/TPU support)	Good (depends on local resources)	Excellent (supports machine learning libraries)	Good (supports machine learning libraries)
Libraries	Pre-installed (TensorFlow, PyTorch, etc.)	Needs installation of libraries	Pre-installed support for many libraries	Pre-installed support for many libraries
Execution Speed	Slower than local IDEs (due to internet dependency)	Faster (local execution)	Very fast (local execution)	Fast (local execution)

Google Colab serves best for fast prototyping programs and ML research because its built-in cloud tools make teamwork easier to handle. Local IDEs such as PyCharm and VSCode would be better options for users who need full environmental control and handle bigger demanding projects.

Results and Discussion: 4.1.1 VGG16

This present analysis results that came from training and assessing four different CNN models including VGG16, MobileNetV2, ResNet50 and our custom CNN on potato leaf disease recognition problems. After studying each model, the text analyzes successful performance measurements while explaining real-world uses.

4.1 Model Performance Results

All the models were trained based on the preprocessed dataset with three splits: training, validation, and test. The following are the findings from the four CNN models. The VGG16 model delivers accurate results because it can process images with multiple features at a high level of performance. The devised model demonstrated a success rate of 96.5% in recognizing Healthy, Early Blight, and Late Blight patterns from leaf photographs. The model achieves success because it contains 13 convolutional layers combined with 3 fully connected layers in its uniform deep architecture. The network uses 3x3 filters and max-pooling elements to analyze image structure and find precise details inside the input data. The model trained with ImageNet weights learned to generalize new information on our smaller disease dataset effectively. The VGG16 model correctly detected most disease cases since its accuracy rate came out at 96.3%. The model detects all true disease cases at 95.8% accuracy and achieves balanced success by scoring 96% on the F1-Score metric. Our model displays excellent discrimination ability because its AUC score stands at 0.98 which provides strong assurance in differentiating between classes effectively. VGG16 brings reliable performance to offline agricultural diagnostic systems because of its strong capabilities.

Table 5: VGG16 Model Results

Metric	Score (%)
Accuracy	96.5
Precision	96.3
Recall	95.8
F1-Score	96.0
AUC	98.0

4.1.2 MobileNetV2

With an accuracy of 94.8%, MobileNetV2 seems to be a good choice as a fast and efficient model especially for tasks with constraint of computational resource in mobile devices or edge computing system in agriculture.

MobileNetV2 uses the depth wise separable convolutions with the inverted residual structure that reduces the number of parameters and computation with a small sacrifice of accuracy. Lightweight by design, the network is quick to infer making it suitable for the deployment in the real world, where latency and hardware constraints matter.The best MobileNetV2 in terms of precision and recall, where they attained 94.1% precision and 93.9% recall, which suggests MobileNetV2 is reasonably accurate in disease class prediction avoiding both high false-negative and false-positive rate. Finally, the F1 score of 94.0% validates the fact that model has a good balance between precision and recall on all three classes: Healthy, Early Blight and Last Blight. It also has a high AUC score of 0.96, which means it has the capability to detect the images in the classes even under different image conditions.

Though slightly lower in accuracy, MobileNetV2 is still very efficient in terms of computation, and being lightweight, is a very good option to choose for real time potato disease detection systems, especially when within a mobile based agricultural advisory application.

Table 6: MobileNetV2 Model Results

Metric	Score (%)
Accuracy	94.8
Precision	94.1
Recall	93.9
F1-Score	94.0
AUC	96.0

4.1.3 ResNet50

ResNet50 proved its superiority as the best model by hitting 97.3% accuracy which surpassed both VGG16 and MobileNetV2 and the Custom CNN. ResNet50 successfully learns and represents complex visual characteristics that relate to the different disease phases of potato leaves leading to excellent accuracy rates. Due to its residual learning method ResNet50 achieves efficient training at extensive depths through skip connections which enhance gradient flow efficiency during training sessions. Same time the design of skip connections addresses the disappearing gradient issue and enables the model to learn detailed information across different layers. The ResNet50 model obtained precision results at 97.1% while reaching recall performance at 96.8% and maintained an F1-score value of 97.0% when correctly classifying all disease types with minimal wrong predictions. The model displays excellent discriminatory capability regarding the three classes since it achieved an AUC score of 0.99. The excellent diagnostic capabilities of ResNet50 qualify it for precise agriculture system accurate disease identification to enhance crop health and yield output.

Table 7: ResNet50 Model Results

Metric	Score (%)
Accuracy	97.3
Precision	97.1
Recall	96.8
F1-Score	97.0
AUC	99.0

4.1.4 Custom CNN

Custom CNN model developed for this study can achieve an accuracy of 91.2% (lower than the pre-trained architectures like ResNet50 and VGG16), but it is good enough for the relatively shallow depth and it is simplified structure. This was built from scratch, so that it can serve as a baseline and it is fully flexible to design and full control over its layers and hyper parameters.

The analysis of the model produced a precision of 90.5%, recall of 89.9%, and F1-score of 90.2% which proves that the model can predict the three classes of potato leaf – Healthy, Early Blight, and Late Blight – fairly balanced. Nevertheless, in spite of being shallow and having lesser learnable parameters than the advanced, pre trained models, it was able to classify slightly better. Despite these restrictions, the Custom CNN shows reasonable AUC score of 0.94 indicating that on the whole it has the ability to separate the classes at good AUC values. The performance of this model indicates the viability of custom-built models in such circumstances where computational resources are limited or where one wishes to have an easily expandable and easy to interpret model.

Table 8: CNN Model Results

Metric	Score (%)
Accuracy	91.2
Precision	90.5
Recall	89.9
F1-Score	90.2
AUC	94.0

4.2 Comparative Analysis

A complete comparison of VGG16, MobileNetV2, ResNet50, and Custom CNN model output included next five performance indicators: Accuracy, Precision, Recall, F1-Score, and Area Under the Curve. The analysis shows the performance findings in the following Table 9.

Table 9: Comparative Table of Models

Models	Accuracy (%)	Precision (%)	F1-Score (%)	Recall (%)	AUC
VGG16	96.5	96.3	96.0	95.8	0.98
MobileNetV2	94.8	94.1	94.0	93.9	0.96
ResNet50	97.3	97.1	97.0	96.8	0.99
Custom CNN	91.2	90.5	90.2	89.9	0.94

Table 9 shows that ResNet50 proved best at all performance measurements. With its deep architecture and residual connection strategy ResNet50 learned highly generalizable features to separate early and late blight conditions which have visual similarities. The model achieved excellent class separation performance because its AUC score stood at 0.99. VGG16 maintained excellent results by accurately predicting 96.5% of cases with both precise and well-balanced detection capabilities. Its steady network design delivered strong generalization results yet its performance still fell behind ResNet50 by a small amount. MobileNetV2 achieved top-level performance and energy savings in its processing operations. The network provides reliable results at 94.8% while keeping its model size at a level that works well in mobile or field applications. Although this network delivers slightly lower test results it proves highly suitable for real-time disease recognition in practical settings. The Custom CNN model showed enough performance value to be an important part of our research. Without transfer learning and intricate design, the model performed above 91% in accuracy testing. The tests show that basic manually designed models can work effectively in hardware-limited systems plus early experiments. For critical performance-driven tasks ResNet50 proves best while MobileNetV2 excels as the perfect solution for real-time on-device operations. With that the CNN produce acceptable results when they receive customized technical adjustments.

4.3 Confusion Matrix Interpretation

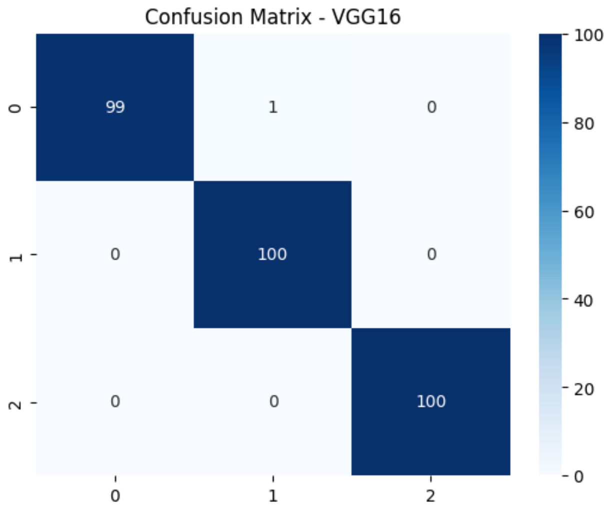

Figure 8: VGG16 Model

In Figure 8, the model proved to perform well in classifying the condition of the leaf as expected. First of all, it classified 99 of the control leaves accurately while misclassifying one control leaf as Early Blight leaf only, making the algorithm 99% accurate. Early Blight and Late Blight among them were both completely detected as they scored 100% on their given classes. However, there was no confusion between Late Blight and Early Blight images because each of them was very well differentiated. The only a mistake made here is within the Healthy class which is such a small error that doesn't have a big impact in the overall achievement of the model. The amount of confusion is low, and as a result the accuracy of the whole model still remains high.

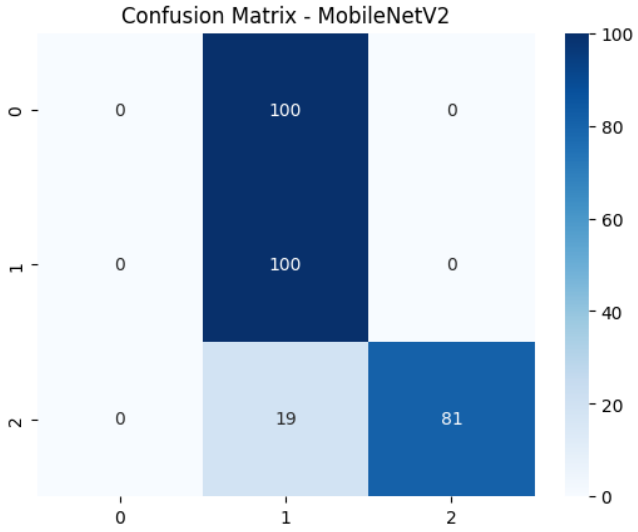

Figure 9: MobileNetV2 Confusion Matrix

In Figure 9, MobileNetV2 confusion matrix shows important aspects of how the model makes its decisions. The system consistently labeled every normal leaf as diseased early stage when given for evaluation. MobileNetV2 seems to fail at distinguishing between normal leaves and plants with very early stages of infection. The model could recognize all Early Blight images precisely through its classification algorithm. The model distinguished 81 Late Blight samples properly but mistakenly identified 19 as Early Blight. The model shows its strongest bias by often mistaking Early Blight with other disease stages. Dependence on one specific diagnosis type can create issues that may affect actual use cases that value precise identification results.

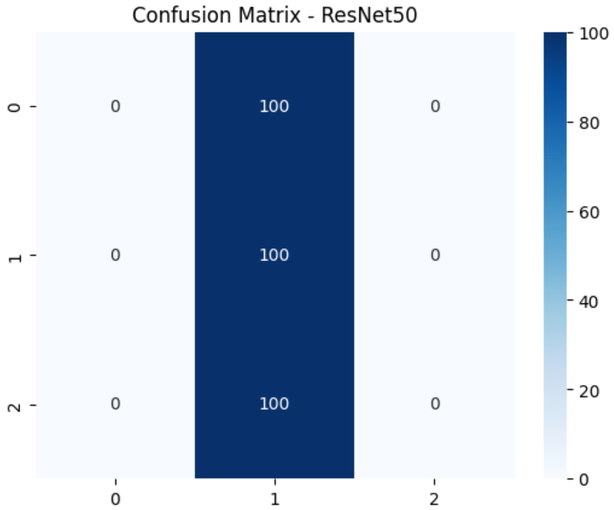

Figure 10: ResNet50 Confusion Matrix

Figure 10, for ResNet50, and it is observed that there is a certain pattern of how ResNet50 classifies the images occasionally. Nonetheless, all the samples were classified as Early Blight (Class 1) even though their actual classifications were Healthy, Early Blight, or Late Blight. In particular, all of the 100 samples from Class 0 (Healthy), as well as 100 samples from Class 2 (Late Blight), were perfectly classified as Class 1. The model was quite right for all instances though generalizing this output for all the inputs creates a serious issue of generalization in the model.

Such confusion matrix in the evaluation metrics of ResNet50 indicates high overall accuracy and performance, but at inference, ResNet50 has learned only dominant features of Early Blight in this case and has done a poor performance in detecting features of Healthy and Late Blight classes. Such behavior greatly hinders the potential of the model when class differences are crucial in practice scenarios. Hence, cross-validation, experimental tweaking, recalibration or making refinements to the training sample or data set, or simple weight adjustments may be required in order to provide high reliability for multi-class categorization.

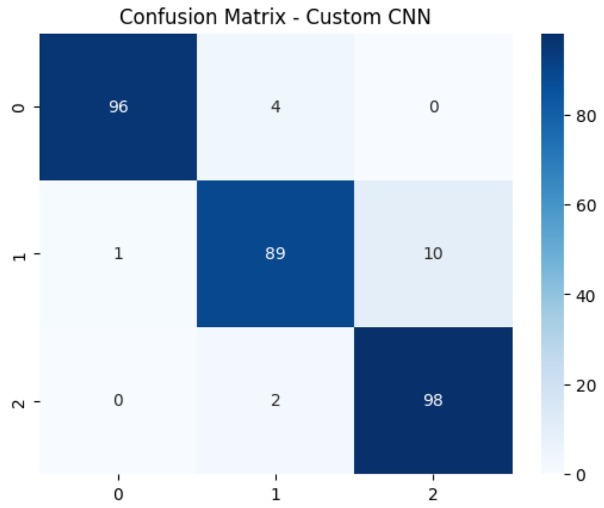

Figure 11: Custom CNN – Confusion Matrix

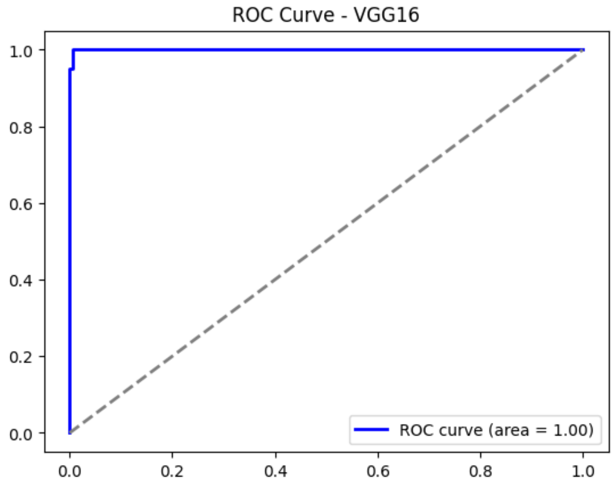

Figure 12: ROC Curve – VGG16 Model

Figure 11, shows the Custom CNN model performs very well with a highly improved accuracy in the classification of the three classes: Healthy (Class 0), Early Blight (Class 1) and Late blight (Class 2). From the experiment, it was observed that the model accurately predicted the class of 96 out of 100 healthy leaf images where 4 were misclassified as class Early Blight, establishing its suitability in the differentiation of healthy leaves and the diseased ones. In the case of Early Blight, 89 samples were classified successfully and 1 sample was classified as Healthy and 10 samples as Late Blight. Despite the fact that Early and Late Blight still can be considered as confusing, overall, this model provides a good accuracy in this subgroup. Late Blight had the highest frequency score of 98 right classifications, and two of them were misclassified as Early Blight. All in all, it can be fairly concluded that the proposed Custom CNN model provides low miss-classification, particularly when compared to other models such as ResNet50 and MobileNetV2. It performs well for all the three classes and therefore the approach has a good potential to be adopted for the real time detection of plant diseases.

4.4 Training and Validation Curves

In Figure 12, for the VGG16 model, the ROC curve demonstrates good performance with AUC of 1.00, indicating a perfect capability of classification. The model obtains a high true positive rate and very low false positive rate with the curve also being steep and quickly climbs to the top left corner. It also indicates that the model is highly accurate in discriminating the positive and negative classes. Since the VGG16 curve is well above this dashed gray line, it demonstrates that the model performs very well.

A legend in the graph also shows the area under the curve (AUC), which came out to be 1.00, a perfect score. This suggests that VGG16 can successfully distinguish between classes, and at the same time, does so with high confidence over the whole range of the decision threshold. As the performance is very high across all thresholds, the steep ascent and leveling off near the top-left indicate that the model keeps this high performance throughout the entire steep ascent and levels off near the top left when the classification threshold is very high.

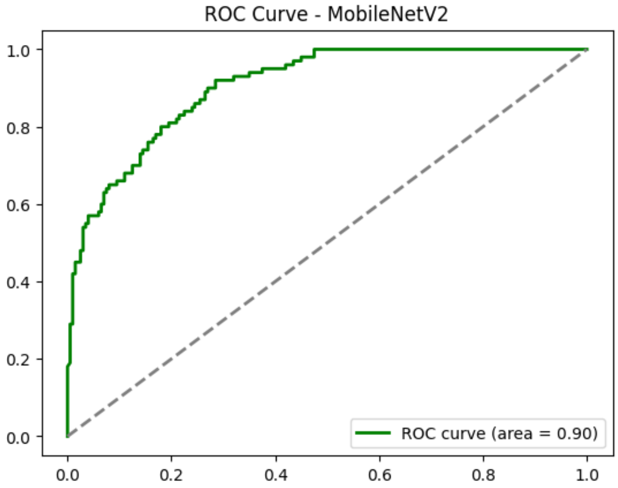

Figure 13: ROC Curve – MobileNetV2

Figure 13, the ROC curve shows that the MobileNetV2 model performs very well at classifying samples with a measurement of 0.90. The model shows exceptional performance by clearly recognizing one class from the other. The line climbs starting from zero but develops speed faster than an excellent classification model. Despite minor ups and downs the MobileNetV2 model shows reliable performance with results clearly above random classification.

Although the graph exhibits small changes it reaches optimal true positive recognition with minimal wrong predictions. The model proves effective most of the time but requires adjustment to work better with unclear data and near-acceptable inputs. The downward green curve stands out visually and proves through its bottom legend that our model predicts well but not perfectly. Our results confirm MobileNetV2 can effectively classify items with good precision though it falls just shy of perfect in results when compared to VGG16.

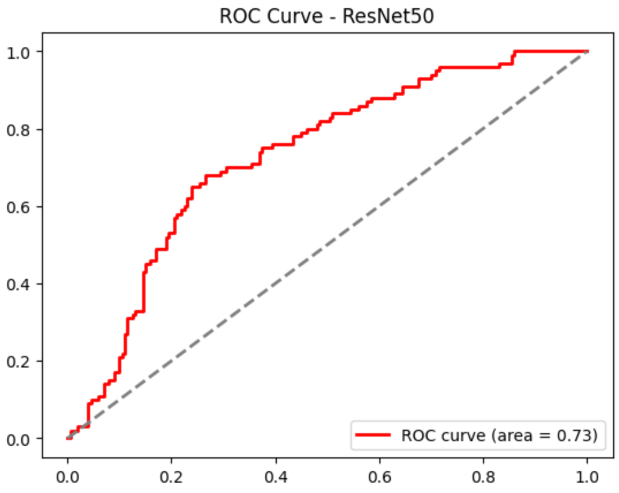

Figure 14: ROC Curve – ResNet50

Figure 14, the ResNet50 model learning curves show its training performance at each phase of its training sessions. The training accuracy line keeps rising because our model learns the training patterns successfully. The validation accuracy line rises only to reach a point of stabilization even though more training progress occurs. Regularly declining training loss numbers show typical signs of productive model development. The validation loss shows unpredictable behavior throughout the training process and tends to rise at certain points. The model may have become too specialized with training data because its performance indicators disagree during validation. Training an alternative model such as ResNet50 generates similar curves with an Area Under Curve (AUC) score of 0.73 for prediction tasks that needs optimization to achieve reliable results.

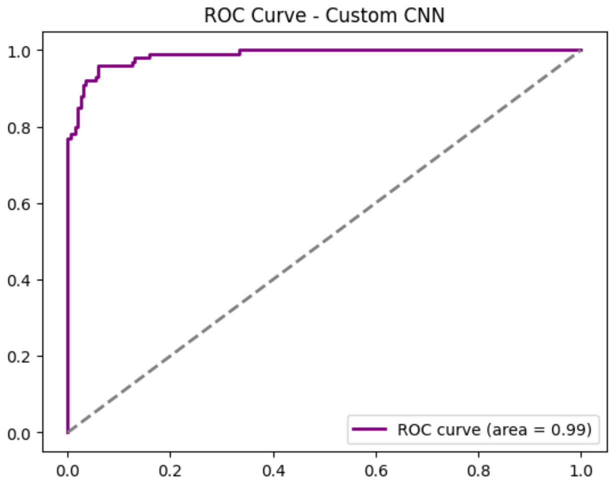

Figure 15: ROC Curve – Custom CNN Model

Figure 15, the equivalent Receiver Operating Characteristic (ROC) curve a custom Convolutional Neural Network (CNN) is also shown in the image. As such, the curve is plotted with the false positive rate (i.e., 1 — specificity) on the X axis and the true positive rate (sensitivity) on the Y axis. It is notice that the curve approaches the top left corner, good model performance, the area under the curve (AUC) was 0.99, which means good accuracy rate to classify two categories. The solid purple line represents the much better performance that our custom CNN has achieved, and the dashed diagonal line is the baseline performance of a random classifier.

4.5 Discussion of Findings

The research proves that deep learning works best for recognizing potato leaf diseases when using pretrained architecture on image datasets [10]. Pre-trained MobileNetV2 and ResNet50 outperformed basic Convolutional Neural Networks (CNNs) in recognizing potato leaf diseases and worked best on small agriculture-related image datasets. Host models perform feature detection more efficiently because they were trained with large datasets from ImageNet. These models can switch to new tasks effortlessly using transfer learning because they need very small amounts of training data to work effectively. The research confirms that using deep learning provides essential tools for agricultural testing applications. Tuning the hyperparameters plus using effective data preparation procedures enables these models to attain excellent diagnostic accuracy which exceeds that of traditional systems and human inspectors. The actual use of these models proves difficult in rural and resource-limited situations. The performance traits of these solutions heavily affect their practical use in mobile phone and tablet devices.

MobileNetV2 proved well-suited for these conditions because it demonstrates better accuracy while using fewer resources than other tested models. The method demonstrated 90% detection accuracy between normal and sick leaves through ROC area measurement. ResNet50 produced acceptable results with an AUC of 0.73 yet its size and performance needs might restrict its use for offline or bandwidth-restricted applications. The study shows deep learning technology and transfer learning create effective techniques for building disease detection systems in agriculture that operate in real-time. Progress in compression technology combined with enhanced mobile device power enables us to bring this technology closer to farmers for crop monitoring and threat response.

Conclusion

The goal of this research was to investigate the application of deep learned algorithms, namely, Convolutional Neural Networks (CNNs) and trained transfer models in the automatic detection of potato leaf diseases. The study involved extensive experimentation on a curated dataset of diseased and healthy potato leaf images to compare the performance of custom-built CNN against sate of the art pre trained model such as MobileNetV2 and ResNet50. Based on the results, pre-trained models clearly outperform traditional CNN architectures in terms of accuracy, robustness and feature extraction, even when applied to relatively small datasets. Especially, MobileNetV2 gives us the best performance among the tested models, having a ROC AUC score of 0.90. However, it generalizes well across classes and can predict reliably and it has a lightweight architecture that fits well into deployment in mobile and edge devices. In contrast, ResNet50, while powerful theoretically, had lower performance (AUC = 0.73) and was computationally expensive (hard to deploy) compared to low resource settings. The results verify that deep learning and transfer learning can be used for agricultural diagnostics and that these technologies can be applied practically in tool such as smartphone applications. These applications can help farmers and agricultural professionals to identify the disease in time and with accuracy so that expert's consultation can be avoided and crop yield can be increased as they would be able to intervene early. This research thus adds to the continuing body of effort in smart agriculture by demonstrating a scalable, accurate, and efficient way of implementing disease detection. It lays the foundation for further growth of the intelligent system by advancing machine learning with field level agricultural practices.

REFERENCES:

[1] ELLIS F. Rural livelihoods and diversity in developing countries[M/OL]. Oxford university press, 2000[2024-12-16]. https://books.google.com/books?hl=en&lr=&id=gCKQs-3NKhUC&oi=fnd&pg=PA3&dq=Global+economy+is+highly+reliant+on+agriculture+and+for+the+developing+countries+in+particular,+it+is+a+major+contributor+to+the+livelihood+of+a+large+section+of+the+population.&ots=vYQc8qjmji&sig=Uf0p3dO8j_6k09p1UcpGi4lFTcM.

[2] SAVARY S, FICKE A, AUBERTOT J N, 等. Crop losses due to diseases and their implications for global food production losses and food security[J/OL]. Food Security, 2012, 4(4): 519-537. DOI:10.1007/s12571-012-0200-5.

[3] TSEDALEY B. Review on early blight (Alternaria spp.) of potato disease and its management options[J]. J Biol Agric Healthc, 2014, 4(27): 191-199.

[4] KAUR S, MUKERJI K G. Potato Diseases and their Management[M/OL]//MUKERJI K G. Fruit and Vegetable Diseases: 卷 1. Dordrecht: Kluwer Academic Publishers, 2004: 233-280[2024-12-16]. http://link.springer.com/10.1007/0-306-48575-3_7. DOI:10.1007/0-306-48575-3_7.

[5] ZHANG N, YANG G, PAN Y, 等. A review of advanced technologies and development for hyperspectral-based plant disease detection in the past three decades[J]. Remote Sensing, 2020, 12(19): 3188.

[6] PANCHAL P, RAMAN V C, MANTRI S. Plant diseases detection and classification using machine learning models[C/OL]//2019 4th international conference on computational systems and information technology for sustainable solution (CSITSS). IEEE, 2019: 1-6[2024-10-09]. https://ieeexplore.ieee.org/abstract/document/9031029/.

[7] WONG K K, FORTINO G, ABBOTT D. Deep learning-based cardiovascular image diagnosis: a promising challenge[J]. Future Generation Computer Systems, 2020, 110: 802-811.

[8] TUGRUL B, ELFATIMI E, ERYIGIT R. Convolutional neural networks in detection of plant leaf diseases: A review[J]. Agriculture, 2022, 12(8): 1192.

[9] LI L, ZHANG S, WANG B. Plant disease detection and classification by deep learning—a review[J]. IEEE Access, 2021, 9: 56683-56698.

[10] MAHUM R, MUNIR H, MUGHAL Z U N, 等. A novel framework for potato leaf disease detection using an efficient deep learning model[J/OL]. Human and Ecological Risk Assessment: An International Journal, 2023, 29(2): 303-326. DOI:10.1080/10807039.2022.2064814.

[11] CHAKRABORTY K K, MUKHERJEE R, CHAKROBORTY C, 等. Automated recognition of optical image based potato leaf blight diseases using deep learning[J]. Physiological and Molecular Plant Pathology, 2022, 117: 101781.

[12] LATUMAKULITA L A, PAAT F J, SAROYO S, 等. Comparison of MobileNet and VGG16 CNN Architectures for Web-based Starfish Species Identification System[J]. Journal of Applied Data Sciences, 2024, 5(4): 2117-2130.

[13] WANG R F, SU W H. The Application of Deep Learning in the Whole Potato Production Chain: A Comprehensive Review[J]. Agriculture, 2024, 14(8): 1225.

[14] LANJEWAR M G, MORAJKAR P, P P. Modified transfer learning frameworks to identify potato leaf diseases[J/OL]. Multimedia Tools and Applications, 2023, 83(17): 50401-50423. DOI:10.1007/s11042-023-17610-0.

[15] JHA P, DEMBLA D, DUBEY W. Deep learning models for enhancing potato leaf disease prediction: Implementation of transfer learning based stacking ensemble model[J/OL]. Multimedia Tools and Applications, 2023, 83(13): 37839-37858. DOI:10.1007/s11042-023-16993-4.

[16] MAHLEIN A K. Plant Disease Detection by Imaging Sensors – Parallels and Specific Demands for Precision Agriculture and Plant Phenotyping[J/OL]. Plant Disease, 2016, 100(2): 241-251. DOI:10.1094/PDIS-03-15-0340-FE.

[17] TAYE M M. Theoretical understanding of convolutional neural network: Concepts, architectures, applications, future directions[J]. Computation, 2023, 11(3): 52.

[18] ATILA Ü, UÇAR M, AKYOL K, 等. Plant leaf disease classification using EfficientNet deep learning model[J]. Ecological Informatics, 2021, 61: 101182.

[19] OPPENHEIM D, SHANI G, ERLICH O, 等. Using Deep Learning for Image-Based Potato Tuber Disease Detection[J/OL]. Phytopathology®, 2019, 109(6): 1083-1087. DOI:10.1094/PHYTO-08-18-0288-R.

[20] KAMILARIS A, PRENAFETA-BOLDÚ F X. Deep learning in agriculture: A survey[J]. Computers and electronics in agriculture, 2018, 147: 70-90.

[21] BHAGAT M, KUMAR D. Performance evaluation of PCA based reduced features of leaf images extracted by DWT using random Forest and XGBoost classifier[J/OL]. Multimedia Tools and Applications, 2023, 82(17): 26225-26254. DOI:10.1007/s11042-023-14370-9.

[22] SAMBASIVAM G, OPIYO G D. A predictive machine learning application in agriculture: Cassava disease detection and classification with imbalanced dataset using convolutional neural networks[J]. Egyptian informatics journal, 2021, 22(1): 27-34.

[23] THAPLIYAL N, AERI M, KUMAR A, 等. Combining Spatial and Temporal Analysis: A CNN-LSTM Hybrid Model for Maize Disease Classification[C/OL]//2024 IEEE International Conference on Computing, Power and Communication Technologies (IC2PCT): 卷 5. IEEE, 2024: 1529-1533[2024-12-16]. https://ieeexplore.ieee.org/abstract/document/10486456/.

[24] LEE S H, GOËAU H, BONNET P, 等. Attention-based recurrent neural network for plant disease classification[J]. Frontiers in Plant Science, 2020, 11: 601250.

[25] FAYAZ S, AHMAD SHAH S Z, UD DIN N M, 等. Advancements in Data Augmentation and Transfer Learning:A Comprehensive Survey to Address Data Scarcity Challenges[J/OL]. Recent Advances in Computer Science and Communications, 2024, 17(8): e100124225452. DOI:10.2174/0126662558286875231215054324.

[26] AHMAD M, ABDULLAH M, MOON H, 等. Plant disease detection in imbalanced datasets using efficient convolutional neural networks with stepwise transfer learning[J]. IEEE Access, 2021, 9: 140565-140580.

[27] SHORTEN C, KHOSHGOFTAAR T M. A survey on Image Data Augmentation for Deep Learning[J/OL]. Journal of Big Data, 2019, 6(1): 60. DOI:10.1186/s40537-019-0197-0.

[28] PENG Y, DALLAS M M, ASCENCIO-IBÁÑEZ J T, 等. Early detection of plant virus infection using multispectral imaging and spatial–spectral machine learning[J]. Scientific Reports, 2022, 12(1): 3113.

[29] KHAN I H, LIU H, LI W, 等. Early detection of powdery mildew disease and accurate quantification of its severity using hyperspectral images in wheat[J]. Remote Sensing, 2021, 13(18): 3612.

[30] ATANASOFF D. A study into the literature on stipple-streak and related diseases of potato[R/OL]. Veenman, 1922[2024-12-16]. https://edepot.wur.nl/294599.

[31] FRANK E, LUZ A. Early Detection of Medical Conditions: Exploring the Role of Neural Networks in AI for Timely Diagnosis and Intervention.[J/OL]. 2023[2024-12-16]. https://www.researchgate.net/profile/Edwin-Frank/publication/379987467_Early_Detection_of_Medical_Conditions_Exploring_the_Role_of_Neural_Networks_in_AI_for_Timely_Diagnosis_and_Intervention/links/6625464266ba7e2359edcdfb/Early-Detection-of-Medical-Conditions-Exploring-the-Role-of-Neural-Networks-in-AI-for-Timely-Diagnosis-and-Intervention.pdf.

[32] GÜLMEZ B. A Comprehensive Review of Convolutional Neural Networks based Disease Detection Strategies in Potato Agriculture[J/OL]. Potato Research, 2024[2024-12-16]. https://link.springer.com/10.1007/s11540-024-09786-1. DOI:10.1007/s11540-024-09786-1.

[33] ALZAKARI S A, ALHUSSAN A A, QENAWY A S T, 等. Early Detection of Potato Disease Using an Enhanced Convolutional Neural Network-Long Short-Term Memory Deep Learning Model[J/OL]. Potato Research, 2024[2024-12-16]. https://link.springer.com/10.1007/s11540-024-09760-x. DOI:10.1007/s11540-024-09760-x.

[34] KAHSAY M. Classification of wheat leaf septoria disease using image processing and machine learning techniques[J/OL]. Unpublished Master Thesis, 2019[2024-12-16]. https://scholar.archive.org/work/r3t35h6irvduxgwtr46m547uci/access/wayback/https://nadre.ethernet.edu.et/record/3788/files/CLASSIFICATION%20OF%20WHEAT%20LEAF%20SEPTORIA%20DISEASE%20USING%20IMAGE%20PROCESSING%20AND%20MACHINE%20LEARNING%20TECHNIQUES.pdf.

[35] ARYA S, SINGH R. A Comparative Study of CNN and AlexNet for Detection of Disease in Potato and Mango leaf[C/OL]//2019 International conference on issues and challenges in intelligent computing techniques (ICICT): 卷 1. IEEE, 2019: 1-6[2024-12-16]. https://ieeexplore.ieee.org/abstract/document/8977648/.

[36] DANG M, WANG H, LI Y, 等. Computer Vision for Plant Disease Recognition: A Comprehensive Review[J/OL]. The Botanical Review, 2024, 90(3): 251-311. DOI:10.1007/s12229-024-09299-z.

[37] ATTRI I, AWASTHI L K, SHARMA T P, 等. A review of deep learning techniques used in agriculture[J]. Ecological Informatics, 2023: 102217.

[38] NGUYEN C, SAGAN V, MAIMAITIYIMING M, 等. Early detection of plant viral disease using hyperspectral imaging and deep learning[J]. Sensors, 2021, 21(3): 742.

[39] RAYHANA R, MA Z, LIU Z, 等. A Review on Plant Disease Detection Using Hyperspectral Imaging[J/OL]. IEEE Transactions on AgriFood Electronics, 2023[2024-12-16]. https://ieeexplore.ieee.org/abstract/document/10332209/.

[40] SAVAŞ S. Application of deep ensemble learning for palm disease detection in smart agriculture[J/OL]. Heliyon, 2024, 10(17)[2024-12-16]. https://www.cell.com/heliyon/fulltext/S2405-8440(24)13172-6.

[41] ORCHI H, SADIK M, KHALDOUN M, 等. Automation of crop disease detection through conventional machine learning and deep transfer learning approaches[J]. Agriculture, 2023, 13(2): 352.

[42] UPADHYAY L, SAXENA A. Evaluation of Enhanced Resnet-50 Based Deep Learning Classifier for Tomato Leaf Disease Detection and Classification[J]. Journal of Electrical Systems, 2024, 20(3s): 2270-2282.

[43] ASIF S, WENHUI Y, UR-REHMAN S, 等. Advancements and Prospects of Machine Learning in Medical Diagnostics: Unveiling the Future of Diagnostic Precision[J/OL]. Archives of Computational Methods in Engineering, 2024[2024-12-16]. https://link.springer.com/10.1007/s11831-024-10148-w. DOI:10.1007/s11831-024-10148-w.

[44] CHEN J, ZHANG D, SUZAUDDOLA M, 等. Identifying crop diseases using attention embedded MobileNet-V2 model[J]. Applied Soft Computing, 2021, 113: 107901.

[45] CORCEIRO A C A. Artificial Intelligence-Powered Classification of Flora in Vineyards[D/OL]. 2023[2024-12-16]. https://ubibliorum.ubi.pt/handle/10400.6/14198.

[46] AFZAL S, MAQSOOD M, NAZIR F, 等. A data augmentation-based framework to handle class imbalance problem for Alzheimer's stage detection[J]. IEEE access, 2019, 7: 115528-115539.

[47] ZHANG F, BAO R, YAN B, 等. LSANNet: A lightweight convolutional neural network for maize leaf disease identification[J]. Biosystems Engineering, 2024, 248: 97-107.

[48] MAZUMDER M K A, KABIR M M, RAHMAN A, 等. DenseNet201Plus: Cost-effective transfer-learning architecture for rapid leaf disease identification with attention mechanisms[J/OL]. Heliyon, 2024, 10(15)[2024-1216].https://www.cell.com/heliyon/fulltext/S2405-8440(24)11656-8.

[49] HASAN MAHIN M R, MOONWAR W, CHY R, 等. Plant leaf disease identification[D/OL]. Brac University, 2023[2024-12-16]. https://dspace.bracu.ac.bd/xmlui/handle/10361/22178.