ISSN: 3078-5367 **DOI:** https://doi.org/10.69513/jncs.v3.i1.a4

Al-Noor Journal for Information Technology and Cybersecurity

https://jncs.alnoor.edu.iq/

A Lightweight Parallel CNN Framework for Real-Time Plant Disease Detection in Corn, Tomato, and Apple

[1]Shamam S. Ahmed, [1]Mamoon A. Al-Jbaar and [1]Emad A. Al-Sabawi

[1] Department of Computer Engineering, Nineveh University, Ninevah 41001, Iraq.

Article information

Article history:
Received: 27-3-2026
Revised: 26-4-2026
Accepted: 28-5-2026

Keywords:
Plant Leaf Disease Detection
Deep Learning
Parallel Convolutional Neural Network (Parallel-CNN)
Binary Classification
Smart Agriculture
Computer Vision

Correspondence:

Shamam S. Ahmed
email:
shamam.shihab.eng23@stu.uoninevah.edu.iq

Abstract

Plant leaf diseases present significant risks to agricultural productivity. The process for diagnosing plant leaf diseases involves the examination of leaves by experts, but this method is quite inefficient and inaccurate. This study presents a novel parallel scheme architecture of deep learning networks, which is based on parallel operation of individual binary classifiers, wherein each classifier is dedicated to identifying a particular disease. This architecture improves feature extraction while keeping low computational complexity and ensures reliability and portability of the model to be utilized on lightweight platforms such as mobile phones. Moreover, the proposed architecture is flexible and extensible to use new classifiers for particular diseases without adding any complexity. The model was trained with a part of the Plant Village data set that comprised images of corn, tomato, and apple crops, each with three types of diseases and a healthy one, with 513 images per category. Preprocessing techniques and data augmentation have been used to enhance generalization. This model has just 710,786 parameters and consumes 1.327 GFLOPs. Field images collected from nature were used to check the robustness of the model; however, environmental conditions and the scarcity of plants made it impossible to capture images of only 3–4 samples per crop through a mobile phone. This reason makes the evaluation preliminary. The accuracy by Parallel-CNN are: corn-Cercospora Leaf Spot 98.54%, Common Rust 100%, Northern Leaf Blight 100%; tomato-Bacterial Spot 100%, Early Blight 100%, Late Blight 98.54%; and apple-Apple Scab 95.63%, Black Rot 100%, and Cedar Apple Rust 100%. The outcome verifies that the proposed model offers an effective, scalable, and practical solution for real-time monitoring of crops that assists in smart agriculture.

DOI: https://doi.org/10.69513/jncs.v3.i1.a4

1. Introduction

Consumerism means using digital technology more sustainably (digital sharing applications, energy efficiency) where digital technology itself impacts (hosting the digital environment, less data) and relies on weak digital forces, including control of the economy and responsibility in industry for a low-cost digital world.

The major challenges that affect plant growth are plant diseases, which can reduce the quality and quantity of crops if not treated. These diseases can destroy crops, thus leading to economic problems [1,2]. Plant diseases are usually diagnosed by visual inspection by experts trained for this task. However, this takes a great deal of time and effort and is prone to errors [2]. To overcome these limitations, the development of technology, artificial intelligence,

ISSN: 3078-5367 **DOI:** https://doi.org/10.69513/jncs.v3.i1.a4

and its entry into most fields have made its use in the field of plant image processing an urgent necessity to detect plant diseases and classify them and solve all problems related to them by analyzing images of plant leaves [1,3], for these technologies can identify precise disease symptoms that the human eye may not notice, which contributes to early and accurate diagnosis. This leads to immediate treatment by farmers at the appropriate time, which reduces crop loss and increases production quality. Among these approaches, Deep learning, especially convolutional neural networks, has played an effective role in this field, as they have proven their efficiency and effective results in the extraction of complex characteristics of images, for these networks simulate the ability of the human brain to process images, and this makes them an ideal solution to the problem of classification of diseases. On the other hand, the performance of deep neural networks often requires high computational resources, which limits their use in resource-constrained rural areas, which is considered one of the most important challenges facing the solution of this problem. Moreover, many state-of-the-art lightweight networks fail to achieve efficiency and accuracy simultaneously while also failing to provide scalable frameworks for learning features specific to diseases. However, most existing lightweight CNN-based approaches focus on single-stream or multi-class classification, which limits flexibility in handling disease-specific learning and real-world deployment scenarios. In addition, limited work has addressed efficient parallel binary classification frameworks that can be extended to multiple crops while maintaining computational efficiency. This necessitates the development of a highly efficient, accurate, and scalable framework for plant disease detection. The key innovation of this work lies in introducing a lightweight parallel architecture based on independent disease-specific binary classifiers, which enables specialized feature learning while reducing computational complexity compared to conventional multi-class CNN models. In addition, unlike existing lightweight architectures, the proposed design provides a scalable framework that allows new disease classes to be incorporated without modifying the overall network structure. The main contributions of this work can be summarized as follows: (i) a lightweight parallel CNN architecture based on independent binary classifiers, (ii) an efficient and scalable design capable of handling multiple disease-specific learning branches, (iii) a cross-platform evaluation framework considering computational efficiency, and (iv) a lightweight deployment-oriented model suitable for embedded and resource-constrained environments. As a consequence, this study brings a research effort on lightweight architecture of Convolutional Neural Networks for plant disease detection. It is aimed at developing an accurate and efficient approach with low computational cost for real-time detection and classification of plant diseases. In particular, performance has been our priority without any deterioration in terms of accuracy. Through proper configuration of the layers of the network, it was possible to design a model that could easily be implemented on low-power platforms; hence the capability to detect plant diseases in real-time is made possible. The outcome of this will be improved agricultural development and economic security for farmers. Farmers living in rural areas will now have access to this technology in an efficient manner. The rest of this paper is structured as follows: Section 2 gives the literature review, Section 3 describes the methodology used, Section 4 discusses the results of experiments conducted, and Section 5 contains conclusions.

2. Related Works

It is crucial to correctly and promptly detect diseases in plants because it ensures improved productivity of agricultural crops and helps avoid any financial losses. Recent advancements in deep learning models, especially CNNs, have proven to work excellently in detecting diseases in plants. The reason behind this is that hierarchical features like colors, texture, and changes in leaf shape can be learned using deep learning models. One of the most adapted methodologies has been transfer learning, coping with limitations in small or imbalanced datasets where some pre-trained models like EfficientNet, VGG, and ResNet are fine-tuned on crop-specific datasets [4,5]. Ahmad et al. (2021) performed stepwise transfer learning addressing class imbalance and reported an improved accuracy from 82.4% to 91.3% [4]. Wang (2022) reported an accuracy of 94.7% in crop disease detection through CNN optimization for feature extraction [5]. However, several challenges arise when binary models generalize the complex field conditions due to occlusion, variable illumination, and overlapping leaves [6,7]. These studies are summarized in Table 1.

Table 1: Binary Classification Studies

Author(s)	Year	Method	Accuracy (%)	Notes
Ahmad et al.	2021	Stepwise Transfer Learning + CNN	91.3	Addressed dataset imbalance
Wang	2022	CNN for crop disease detection	94.7	Optimized feature extraction

ISSN: 3078-5367 **DOI:** https://doi.org/10.69513/jncs.v3.i1.a4

Sun et al.	2020	CNN for Northern maize leaf blight	87.9	Tested under low illumination and occlusion
Fuentes et al.	2020	CNN under field conditions	92.5	Considered environmental variability

Deep CNNs perform well in terms of discrimination of different disease varieties in multiple classes. Sun et al. (2020 study focuses on binary/field maize disease classification, while Sun et al. (2025) refers to a different multi-class EfficientNet-based study on corn leaf disease) [8] applied EfficientNet to corn leaf disease, with a reported accuracy of 96.3%, while Bleasdale & Whyatt (2025 journal article supported by a bioRxiv preprint and dataset release; preprint and dataset should be explicitly distinguished from the peer-reviewed publication) [9] used multispectral imagery coupled with CNN to detect apple scab at an accuracy of 98%.The main shortcomings are limited labeled data across all disease categories, low-resolution images of the fields, and variability in environmental conditions that may lead to lower generalization. The multi-class studies have been summarized in Table 2.

It should be emphasized that Sun et al. (2020) considers the problem of detecting maize diseases based on binary classification under field conditions, while Sun et al. (2025) discusses the problem of classifying diseases of corn leaves through multi-classification via EfficientNet. The two papers differ significantly in terms of task formulation and experimental conditions.

Table 2: Multi-Class Classification Studies

Author(s)	Year	Method	Accuracy (%)	Notes
Sun et al.	2025	EfficientNet for corn leaf disease	96.3	High accuracy classification (distinct from Sun et al. 2020 binary study)
Bleasdale & Whyatt	2025	Multispectral + CNN	98	Early apple scab detection (peer-reviewed journal publication supported by a bioRxiv preprint and a DataCite-registered dataset)
Paul et al.	2023	Real-time CNN for tomato disease	95.6	Mobile deployment
Yulita et al.	2023	Dense CNN for real-time tomato detection	93.4	Field applicability
Khan et al.	2024	Deep Transfer Learning for corn	96.1	Fine-grained disease classification
Siam et al.	2024	SE-VGG16 MaizeNet	97.2	Attention-based maize detection

Crop-specific studies further explain the flexibility of CNNs and transfer learning. In the case of corn, the paper proposed by Siam et al. (2024) using SE-VGG16 MaizeNet attained 97.2% [13], whereas Khan et al. (2024) used fine-grained transfer learning and attained an accuracy rate of 96.1% [12]. On the other hand, Ahadian et al. (2024) utilized a weighted loss in CNNs for solving the problem of class imbalance and got an accuracy of 95.3% [14]. In the case of tomato diseases, Paul et al. (2023) presented a real-time CNN-based approach for mobile applications [10], Wu et al. (2021) recorded 94.1% using deep CNNs [15], and Yulita et al. (2023) reached 93.4% for classification in a real-time field [11]. For apple leaf disease classification, Jiang et al. (2019) attained a recognition accuracy of 95.2% [16], while Bansal et al. (2021) achieved an accuracy of 94.5% [17]; Ozden (2021), on the other

ISSN: 3078-5367 **DOI:** https://doi.org/10.69513/jncs.v3.i1.a4

hand, attained an accuracy of 91.6% [18] and Assad et al. (2023) an accuracy of 93.9% by used EfficientNet [19]. Some of the main difficulties that arise include environmental variability, low-resolution imaging, class imbalance within datasets, and noisy field data. Crop-specific studies have been listed out in Table 3 below.

Table 3: Crop-Specific Disease Classification Studies

Crop	Author(s)	Year	Technique	Accuracy (%)	Notes
Corn	Siam et al.	2024	SE-VGG16 MaizeNet	97.2	Real-world deployment
Corn	Khan et al.	2024	Deep Transfer Learning	96.1	Fine-grained classification
Corn	Ahadian et al.	2024	Weighted-loss CNN	95.3	Addressed class imbalance
Tomato	Paul et al.	2023	Real-time CNN	95.6	Mobile app deployment
Tomato	Yulita et al.	2023	Dense CNN	93.4	Real-time field detection
Apple	Assad et al.	2023	Transfer Learning + EfficientNet	93.9	Large-scale dataset adaptation
Apple	Bansal et al.	2021	Deep CNN	94.5	High-resolution datasets
Apple	Ozden	2021	Transfer Learning	91.6	Domain-specific fine-tuning

Dataset collection is crucial to build the strongest models. The datasets obtained under controlled conditions using PlantVillage serve as ground-truth datasets with great precision [1,2,3]. Nevertheless, the field-collected datasets improve generalization in real-world situations, introducing variations due to illumination, occlusion, and diverse environments [20,21,16]. Datasets with multispectral imagery can detect symptoms early through invisible wavelengths [9,22]; however, such datasets require sophisticated hardware and complicated preprocessing techniques. The types of datasets used for this research are summarized in Table 4.

Table 4: Performance on Plant Disease Datasets

Dataset Type	Author(s)	Year	Model	Accuracy (%)	Notes
PlantVillage	Zheng et al.	2019	Transfer Learning + CNN	97	Controlled dataset
PlantVillage	Mohameth et al.	2020	CNN	95	Benchmark dataset
Field-Collected	Deng et al.	2023	Mobile-based CNN	94	Real-world field images

ISSN: 3078-5367 **DOI:** https://doi.org/10.69513/jncs.v3.i1.a4

Field-Collected	Mallick et al.	2023	CNN	93	Mung bean pest detection
Multispectral	Bleasdale & Whyatt	2025	CNN	98	Early apple scab detection (peer-reviewed journal publication supported by a bioRxiv preprint and a DataCite-registered dataset)
Multispectral	Chen et al.	2021	CNN	96	Precision agriculture applications

However, the overall performance of deep learning algorithms is highly promising when applied to plant disease diagnosis, irrespective of the type of crops being used or the environment. However, there remain some challenges that include the need for large amounts of labeled data, the presence of poor-quality images, field variations, occlusion, and imbalance between classes. Although hybrid architectures, attention models, and multimodal imaging could help address some of the issues, they also come with increased computational costs and complexity. Unlike the previous studies, where the aspect of computational efficiency was not adequately addressed, this study takes into account the practicality of designing a model that can operate efficiently in edge computing environments, such as agriculture. In comparison to the other studies, the proposed system is designed as a parallel model that comprises multiple binary classifiers as opposed to a singular multi-classifier model. The rationale behind this approach is to enhance computational efficiency and specialization per disease type.It should be emphasized that at present, the model is only capable of handling binary classification tasks per disease type and is unable to handle instances where there are multiple diseases within an image. This is an area for future development.Future work should be the design and implementation of a lightweight, generalizable, and robust model for agricultural purposes, including its extension to multi-label learning, robustness in a real-world scenario, and execution time efficiency. Furthermore, there should be a focus on the evaluation of execution time and efficiency of the system through the gem5 simulation environment.

3. Methodology

3.1 Dataset

A publicly available collection of 54,305 images of healthy and diseased plant leaves called PlantVillage Dataset has been used in this article [23]. This dataset involved 14 crop species, which are apple, blueberry, cherry, grape, orange, peach, pepper, potato, raspberry, soybean, squash, strawberry, and tomato, where these images were captured under controlled conditions. It comprises images of 17 distinct diseases, categorized as 4 bacterial diseases, 2 mold-related diseases (oomycete), 2 viral diseases, and 1 mite-caused disease. Moreover, 12 of the crop species feature images of healthy leaves with no visible disease symptoms. Figure 1 clarifies samples of this dataset for multiple plant leaves.

ISSN: 3078-5367 **DOI:** https://doi.org/10.69513/jncs.v3.i1.a4

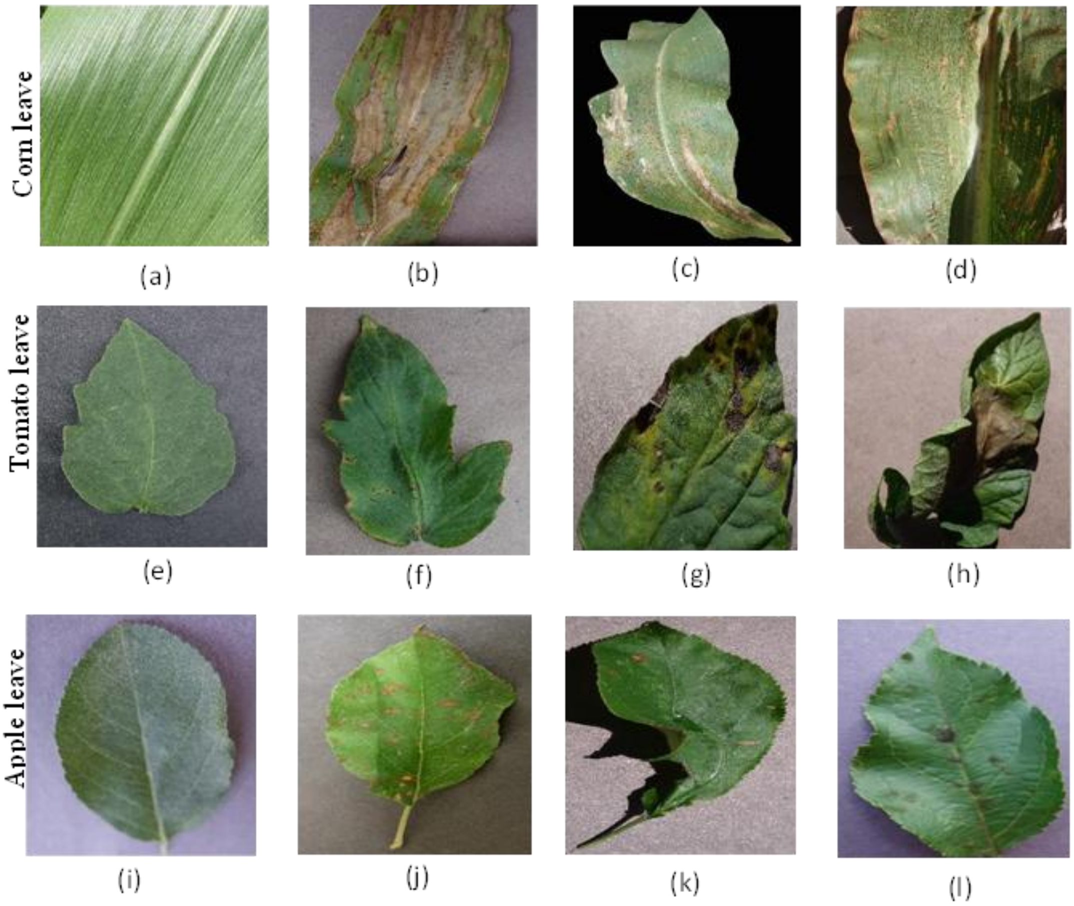


Figure 1: (Corn leaves) (a) Health,(b) Northern_Leaf_Blight,(c) Common_rust, (d) Cercospora_leaf_spot Gray_leaf_spot. (Tomato leaves) (e) Healthy, (f) Bacterial_spot,(g)Early_blight, (h) Late_blight.

(Apple leaves)(i) Healthy, (j) Cedar_apple_rust, (k) Black_rot, (l) Apple_scab.

As illustrated in Figure 1, the images within the database exhibit variations in resolution, lighting intensity, and clarity. Consequently, pre-processing is required before these images can be used as inputs for the deep network. This step is essential to enhance feature extraction, improve network efficiency, and boost overall accuracy. Moreover, to enhance the stability of the training process and provide balanced learning among all categories, a portion of the data set was created such that 513 images were chosen for each category. It is important to note the fact that although the original PlantVillage dataset consisted of 54,305 images, only a few images out of them were selected for this study. In particular, 513 images of each category (corn, tomato, and apple) were collected in order to create an unbiased and evenly balanced dataset.

In order to test the efficiency of the proposed method in a realistic farming environment, an independent testing dataset comprising real leaf images obtained from fields in natural light conditions was used. This dataset represents realistic problems such as uneven illumination, background noise, partial occlusions of leaves, and overlapping of leaves. It is worth mentioning that the external dataset is quite small in size and, therefore, is not appropriate for any kind of training process and can be utilized solely for the assessment of the generalization abilities of the trained model. A set of example field images used for external testing is shown in Figure 2. This external validation approach guarantees that the performance of the model is not constrained only to controlled laboratory data but can be generalized effectively into real-world agricultural applications.

ISSN: 3078-5367 **DOI:** https://doi.org/10.69513/jncs.v3.i1.a4

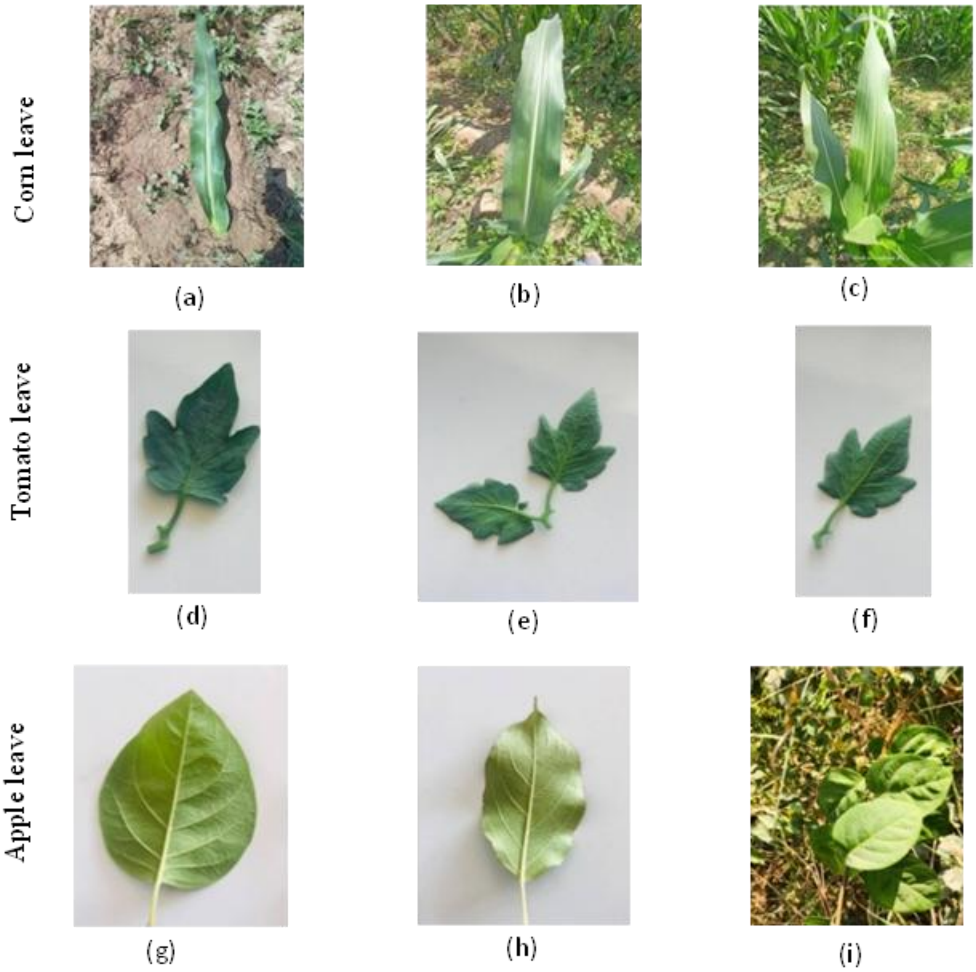


Figure 2: Sample real-world field images used for external testing

3.2 Pre-processing

This stage is considered a significant step towards the development of the classification system, which is based on deep convolutional neural networks. The major goal of this stage is to process the input data in a consistent and reliable manner, which can guarantee the training process as well as the feature extraction process. In this case, since the dataset, as described in this study , contains images of corn, tomato, and apple leaves, both in healthy and diseased conditions, a number of pre-processing steps were considered to improve the quality of the input data, thereby enhancing the ability of the model to generalize well. The first step was to divide the available dataset into training and testing datasets, with a ratio of 80:20, which ensures a fair and unbiased evaluation of the model. This split was performed using a stratified sampling strategy to preserve the class distribution across both training and testing subsets. No cross-validation was applied; therefore, performance evaluation is based on a fixed stratified 80:20 train-test split. Following this step, several data augmentation techniques were performed on the training dataset, which could add to the variability of the dataset, thus minimizing overfitting along with improving the robustness of the model. It must be noted that the process of data augmentation and preprocessing was carried out solely on the training dataset, while the test dataset remained untouched.

3.2.1 Data Augmentation

Data augmentation is a very powerful method that significantly improves the performance of deep learning networks, especially when employed in classification problems. By artificially creating variations in real-life scenarios, data augmentation increases the generalizability of the model, enlarges the dataset, and improves the overall accuracy of the classification process. Moreover, it prevents overfitting, thus ensuring the reliability of the model. Therefore, in the current paper, we will use data augmentation to maximize the performance of the suggested architecture. Whereas a set of implicit processing operations were performed, each image from the approved database represented rotation within a range of ±30 degrees, up to 30% horizontal shifts, up to 30% vertical shifts, A shear transformation, with up to 30% range, has been applied to introduce geometric distortions. Random zooming was also used by up to 30%, and to expand

ISSN: 3078-5367 **DOI:** https://doi.org/10.69513/jncs.v3.i1.a4

the dataset, random horizontal flipping was used, generating mirrored versions of the images and introducing additional variability. these processes contributed to improving the classification results while reducing the overfitting.

3.2.2 Data Balance

To ensure a fair evaluation of model performance and better representation of minority classes, which helps in creating more precise decision boundaries, we balanced the data across all classes used for training. In this study, corn, tomato, and apple leaves were selected as a case study. The proposed system was trained to detect and identify three diseases for each plant, in addition to recognizing their healthy state. For data balancing purposes, 513 images per class (diseased and healthy) have been chosen. Preprocessing of these images was done and the model was trained using these images. The above data set has been selected from the larger PlantVillage data set, which has guaranteed balanced representation of all classes, thus maintaining stability during the training process. This helps in taking decisions without any kind of discrimination against any class while classifying.

3.3 Proposed deep neural network

In this work, we present a deep neural network with few learning parameters compared to typical deep neural networks, which are usually employed in applications such as diagnosis and classification. Traditional neural networks have a large number of parameters and are hard to train, and they require a lot of hardware resources. The proposed network addresses these limitations by offering a more efficient and resource-friendly solution. The architecture of the proposed network is shown in Figure 3..

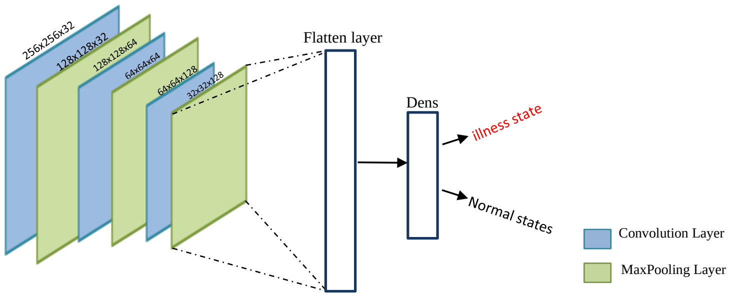


Figure 3.: The proposed Deep Network

As shown in Figure 3., the proposed network consists of only three convolutional layers, which are responsible for extracting the deep features of the target. The most significant features are identified and refined through the three max-pooling layers interspersed between the convolutional layers. The flatten layer then transforms the extracted features into a one-dimensional vector, which is fed into the classifier. The classifier performs binary classification, determining whether the input corresponds to a healthy state or an illness state. To improve the performance and efficiency of the network, we implemented a parallel synchronous structure of deep networks. In this approach, a second network, identical to the first, was integrated to operate in parallel. The features extracted from both networks were combined and fed uniformly into the binary classifier. This parallel structure delivers superior results compared to a large-scale, multi-layer deep network, while maintaining a significantly smaller size and reduced complexity relative to traditional deep networks. The parallel architecture is illustrated in Figure 4..

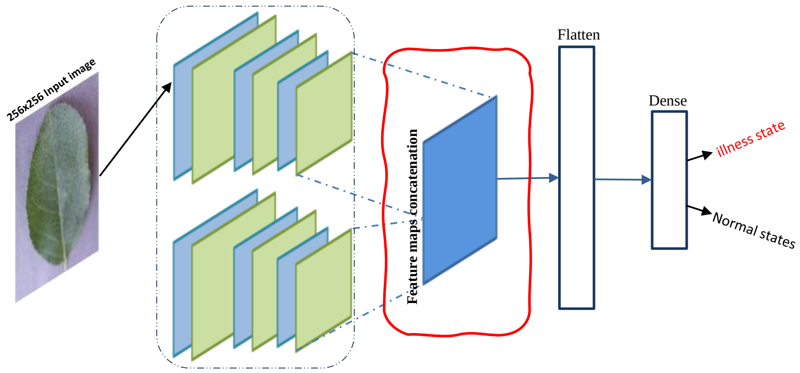


Figure 4.: The parallel deep networks

ISSN: 3078-5367 **DOI:** https://doi.org/10.69513/jncs.v3.i1.a4

Table 5. : Infrastructure of deep parallel networks

Layer (type)	Output shape	Parameters
Input layer	(None, 256, 256, 3)	0
Conv2D_1 (Conv2D)	(None, 256, 256, 32)	896
Conv2D_2 (Conv2D)	(None, 256, 256, 32)	896
MaxPooling2D_1	(None, 128, 256, 32)	0
MaxPooling2D_2	(None, 128, 256, 32)	0
Conv2D_3 (Conv2D)	(None, 128, 128, 64)	18,496
Conv2D_4 (Conv2D)	(None, 128, 128, 64)	18,496
MaxPooling2D_3	(None, 128, 128, 64)	0
MaxPooling2D_4	(None, 128, 128, 64)	0
Conv2D_5 (Conv2D)	(None, 64, 64, 128)	73,856
Conv2D_6 (Conv2D)	(None, 64, 64, 128)	73,856
MaxPooling2D_5	(None, 32, 32, 128)	0
MaxPooling2D_6	(None, 32, 32, 128)	0

In the parallel architecture illustrated in Figure 4. and Table 5., features are extracted simultaneously from both networks, with the most significant ones being selected at each stage. This process culminates in the fusion of the chosen features from the two networks, providing the binary classifier with dual inputs that enhance decision-making accuracy. The feature fusion in this work is implemented using concatenation along the channel dimension, which preserves all extracted features from both networks without information loss. Equations (1) and (2) provide the mathematical representation of the convolution operation applied independently in both branches of the proposed parallel network.

Final feature map of deep Net1

Final feature map of deep Net۲

$$\left.\begin{aligned} X'_{abc} &= b_c + \sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{z=1}^{d} K_{ijzc}.\ \ X_{a+i-1,b+j-1,z} \\ X''_{abc} &= b_c + \sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{z=1}^{d} K_{ijzc}.\ \ X_{a+i-1,b+j-1,z} \end{aligned}\right\} \quad ....(1)$$

Where X' , $X''$are the final features maps extracted from both deepNet1 and deepNet2, respectively .

The definitions of variables are as follows:

- X: input feature map
- X', X'': output feature maps of the two parallel networks
- a , b: spatial coordinates of the output feature map (height and width positions)

ISSN: 3078-5367 **DOI:** https://doi.org/10.69513/jncs.v3.i1.a4

- c: index of the output feature map (channel index)
- $i\,,j$: spatial indices of the convolution kernel representing the row and column positions within the filter window
- z: depth index of the input feature map
- n, m: height and width of the convolution kernel
- d: number of input channels
- K_{ijzc}: convolution kernel weights connecting input channel z to output channel ccc
- b_c: bias term corresponding to the c^{th} feature map

The fused feature representation is obtained using channel-wise concatenation as:

$$F = \sigma(X' \| X'') \quad \dots\dots\dots\dots (2)$$

where $\|$ denotes feature concatenation, F represents the fused feature map obtained after concatenation and σ represents the activation function applied after fusion. As evident from the above formulation, the binary classifier is provided with twice the number of features, significantly enhancing its ability to make more accurate and robust decisions.

This parallel design is preferred over simply increasing network depth or width, as it allows learning diverse feature representations while maintaining a lower computational cost and reduced number of parameters. Increasing depth or width would significantly increase model complexity without proportional performance gains.

The primary innovation in this research can be attributed to the creation of a lightweight parallel framework for deep learning based on several small binary CNN classifiers working in parallel through a systematic and scalable architecture. As opposed to traditional deep neural networks or even typical multi-class classifiers, the designed architecture splits the problem into several distinct binary learning processes, where every binary classifier focuses on one particular plant disease by using the same set of features. The other innovation of this research is the two-stage decision process that involves initial binary classification of disease presence and later aggregation of those decisions to mimic multi-class output, but not requiring a multi-class classifier directly.

The system's training process ran for a constant number of 100 epochs in each case. This was chosen following initial tests carried out with different epoch counts, where 100 epochs was determined to be an optimal choice in terms of both convergence and efficiency. This ensures that all systems being analyzed will be trained under similar conditions.

3.4 The parallel deep system for Detecting and classifying Plant diseases

As outlined in the preceding paragraph and illustrated in Figure3.4, two compact deep networks were integrated to detect and diagnose a specific disease condition in plant leaves with exceptional precision. This concept can be extended to simultaneously identify other diseases by incorporating additional parallel networks within a scalable deep learning framework. The proposed approach in this work highlights a key feature of scalability, allowing the system to expand or contract in size based on the specific requirements of the application. This flexibility ensures adaptability to varying diagnostic needs.as shown in Figure 3.5.During the inference process, the same image is concurrently sent through several parallel binary classifiers, with each classifier specializing in the detection of one disease specifically. Each binary classifier gives binary output, either diseased or healthy. The final inference is made by picking the most confident classifier concerning the disease, thus making the system behave like multi-class classification despite employing binary classifiers. Each of the individual binary classifiers runs independent of one another, without sharing any decisional dependencies amongst themselves, making sure that decision-making remains parallel and non-interfering among the disease-related predictions made. This helps in ensuring that no matter how many classifiers the system uses, its performance remains the same. Therefore, the proposed system can be formally described as a parallel binary decision system, where multiple independent binary classifiers operate simultaneously on the same input image, each specialized in detecting a specific disease.

In the process of implementation of this system, different plants are characterized using three binary classifiers each, in which every single classifier recognizes one specific disease affecting the plant. Each classifier receives the input image at once, and training the classifiers ensures that the classifier will only give a result when there is that specific disease.

Therefore, for each plant, the system will give three binary results, but only one classifier out of the three should give an indication of "diseased" based on the disease identified. The other two classifiers will produce a result of "healthy".

ISSN: 3078-5367 **DOI:** https://doi.org/10.69513/jncs.v3.i1.a4

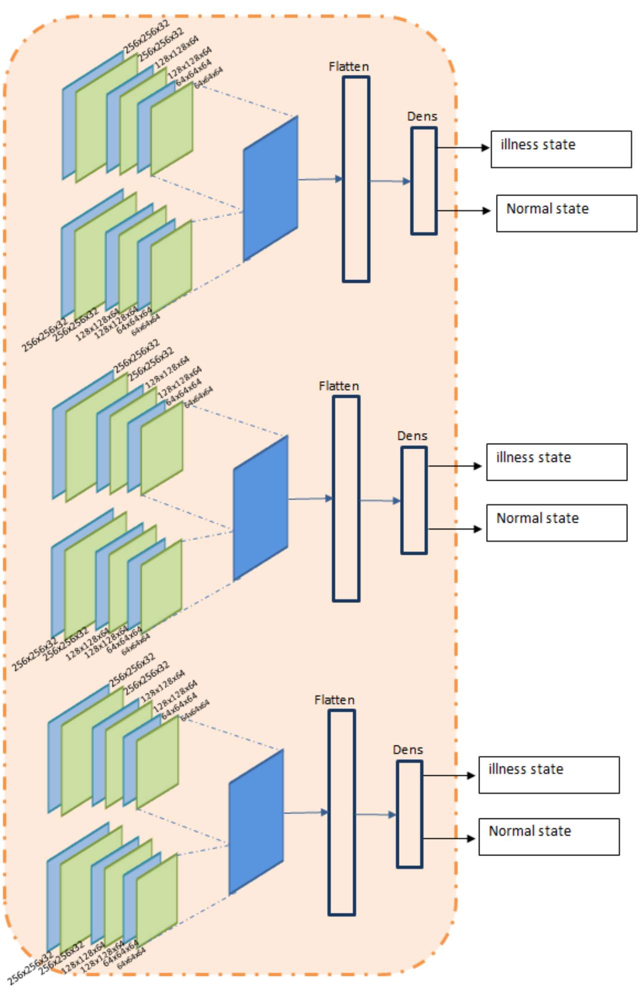


Figure 5.the proposed parallel deep system

As a case study to evaluate the system's performance, corn leaves, tomato leaves, and apple leaves were chosen to diagnose three diseases for each plant, along with their healthy condition, as demonstrated in Table 6.. Each disease is handled by a dedicated binary classifier, where the target disease is labeled as “diseased” while all other conditions are treated as “healthy” for that specific classifier. This design enables specialization and improves classification accuracy.

Table 6.: Plant leaf cases diagnosed by the deep parallel system

Plant Leaf	**Leaf State / Disease**
Corn	Healthy
	Northern Leaf Blight
	Common Rust
	Cercospora Leaf Spot
Tomato	Healthy
	Bacterial Spot

ISSN: 3078-5367 **DOI:** https://doi.org/10.69513/jncs.v3.i1.a4

	Early Blight
	Late Blight
Apple	Healthy
	Cedar Apple Rust
	Black Rot
	Apple Scab

4. Results

4.1 Investigating the Efficiency of Pre-Trained Networks

At the first stage of our research work, testing was done on a set of deep-learning backbones such as VGG16, Inception V3, and Efficient Net. Using the Plant Village dataset to specify which of them is more suitable to emulate. According to Table 7., which summarizes the accuracy evaluation, and the model size of those networks, VGG16 is the suitable one for our application.

Table 7. Effect of Changing the Pre-trained Network Model

Model	**Accuracy**	**Model Size**
VGG16	92.50%	~15 million
InceptionV3	90.62%	~23.8 million
EfficientNet	86.88%	~5.3 million

first, we begin by building a deep network that has the same architecture as VGG16. Gradually adding and removing layers, we watched the accuracy of our network, which was not good enough. Increasing the number of layers made the network bigger with little difference in results, whereas reducing it significantly reduced the accuracy. We also tried changing many other hyperparameters like batch size, learning rate, and number of epochs; after each trial, we analyzed the different performances in order to find out the best option. We also tried several optimizers such as Adam, RMSprop, and SGD in order to improve our model as table . Among the above-mentioned, the best one proved to be the Adam optimizer.

The Table 8.below illustrates the impact of changing the optimizer on the network's performance.

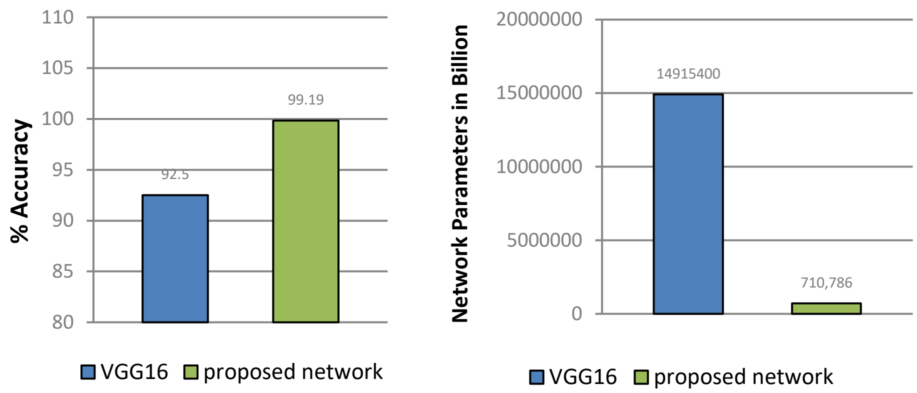
Table 8. The effect of changing the optimizer in proposed-model

Optimizer	**Train Accuracy**	**Validation Accuracy**	**Train Loss**	**Validation Loss**
Adam	0.4238	0.3984	0.5527	0.5540
RMSprop	0.2578	0.2500	0.5558	0.5645
SGD	0.2285	0.2812	0.6926	0.6926

With all the above changes, the overall accuracy remained unsatisfactory. Logically, it seemed time to rethink the way a network should be designed to have higher accuracy with the least model size. Based on the analysis and visualization of each layer during training, a new deep network was designed to simulate VGG16 while significantly reducing its size and

complexity. The comparison between those two networks is illustrated in Figure 6.

ISSN: 3078-5367 **DOI:** https://doi.org/10.69513/jncs.v3.i1.a4



(a) (b)

Figure 6. The proposed Deep Network Compared with the VGG16 Network, (a) Accuracy comparison; (b) Network parameters

This proposed network achieves excellent performance with only 710,786 trainable parameters, obtaining very high accuracy on the three plant types, respectively. The network size is about 0.046 times that of VGG16, representing a 21.7 fold reduction, while maintaining high accuracy.

The computational complexity for VGG16 in terms of GFLOPS for all convolution and dense layers is represented by Equations (4.1) and (4.2), respectively. The proposed compact network reduces the overall computational operations from 15.3467 GFLOPS of VGG16 to 1.3272 GFLOPS. The proposed method approximately reduces the computation by ~ 11.5×, as reflected in Figure 7.

$$\boldsymbol{GFLOP}_{\text{total}} = \sum_{j=0}^{m} \boldsymbol{Ops}(\boldsymbol{j}) + \boldsymbol{D\_Ops} \ldots\ldots(\mathbf{4.1})$$

Where : D_Ops represents the dense layer operations , m is the number of convolutional layers, and Ops(j) is the total operations in the Convolutional layer, calculated as:

$$\boldsymbol{Ops} = \boldsymbol{I}/\boldsymbol{Pch} \times \boldsymbol{O}/\boldsymbol{Pch} \times \boldsymbol{K_H} \times \boldsymbol{K_W} \times \boldsymbol{O}/\boldsymbol{P_H} \times \boldsymbol{O}/\boldsymbol{P_W} \ldots\ldots(\mathbf{4.2})$$

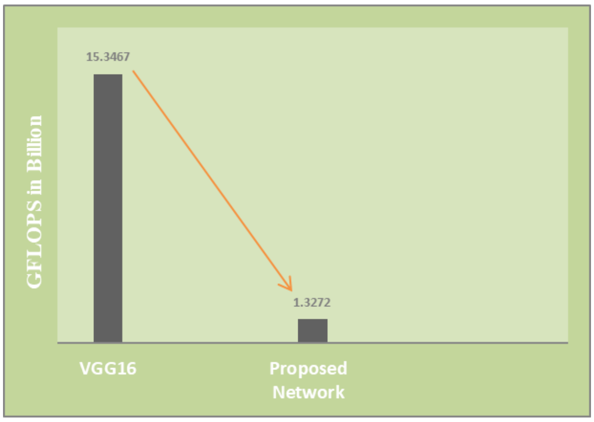


Figure 7: GFLOPS of VGG16 and the Proposed Compact Network

This demonstrates that the network proposed here achieves very high classification accuracy with extremely low computational cost, making it highly suitable for deployment in low-power, real-time agricultural applications.

4.2 Evaluation of the Proposed Lightweight Parallel CNN Model

A deep network following the architectural specification of VGG16 was designed at the outset of this study. This gave a clear baseline, but experiments showed that the addition of more layers significantly increased the model size with minimal improvement in accuracy, while fewer layers meant decreased performance. That being said, there was some enhancement in the results after tuning the parameters and testing different optimizers; however, the overall performance of this model remained unsatisfactory.

To overcome these limitations, the Parallel CNN lightweight approach was proposed. This model contains two small CNN branches which process the same image in parallel, extracting different visual features; their outputs are combined prior to the final classification layer. It improves

ISSN: 3078-5367 **DOI:** https://doi.org/10.69513/jncs.v3.i1.a4

accuracy by incorporating complementary information while keeping the model compact.

To increase the generalization ability, we trained our network using subsets that had an equal amount of data for disease and healthy categories. With 710,786 trainable parameters, this structure gave outstanding performance: 99.84% accuracy for corn, 99.51% for tomatoes, and 98.71% for apples. The obtained results provide an opportunity for us to implement this model into real-time application on small embedded computers, making it possible to use agriculture in various crops.

Equation (4.3) defines the accuracy of the CNN network:

$$\textbf{Accuracy} = \frac{\textbf{TP + TN}}{\textbf{TP+TN+FP+FN}} \text{.......(4.3)}$$

TP stands for True Positive where the model correctly detects the diseased samples, while TN denotes True Negative, where the model correctly detects the healthy samples. In contrast, FP stands for False Positive, which shows healthy samples detected to be diseased, while FN indicates False Negative, representing diseased samples detected to be healthy. Figures 4.3–4.5 and Table 9 summarize the classification accuracy of the proposed parallel CNN model for each plant disease, illustrating its performance across corn, tomato, and apple crops.

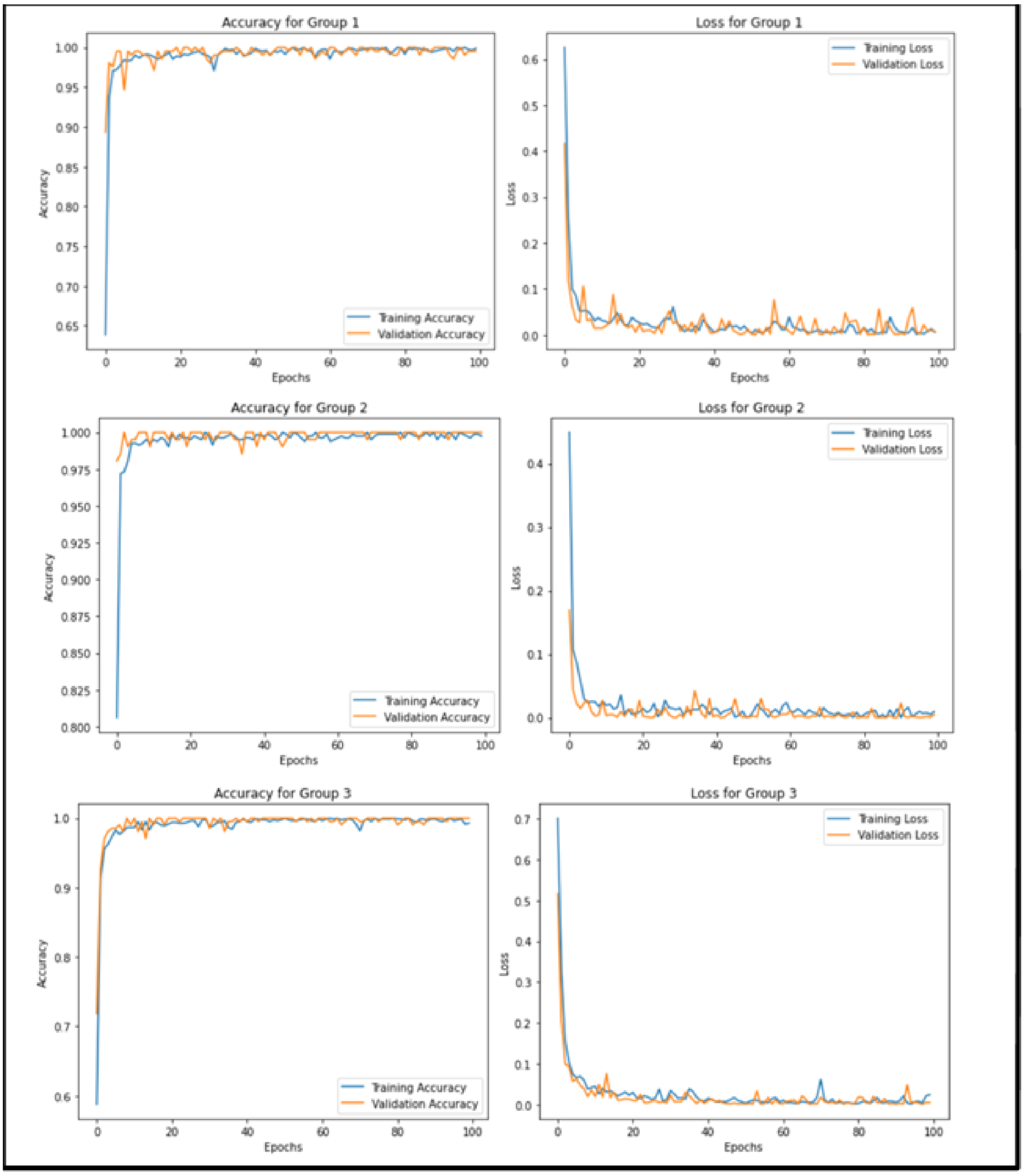


Figure 8: Prediction accuracy of the proposed Parallel CNN model.(a) Model accuracy and loss for Cercospora Leaf Spot vs. Healthy,(b) Model accuracy and loss for Common Rust vs. Healthy,(c) Model accuracy and loss for Northern Leaf Blight vs. Healthy.

A l-Noor Journal for Information Technology and Cybersecurity, Vol.3, No.1, 2026 (27-47)

ISSN: 3078-5367 **DOI:** https://doi.org/10.69513/jncs.v3.i1.a4

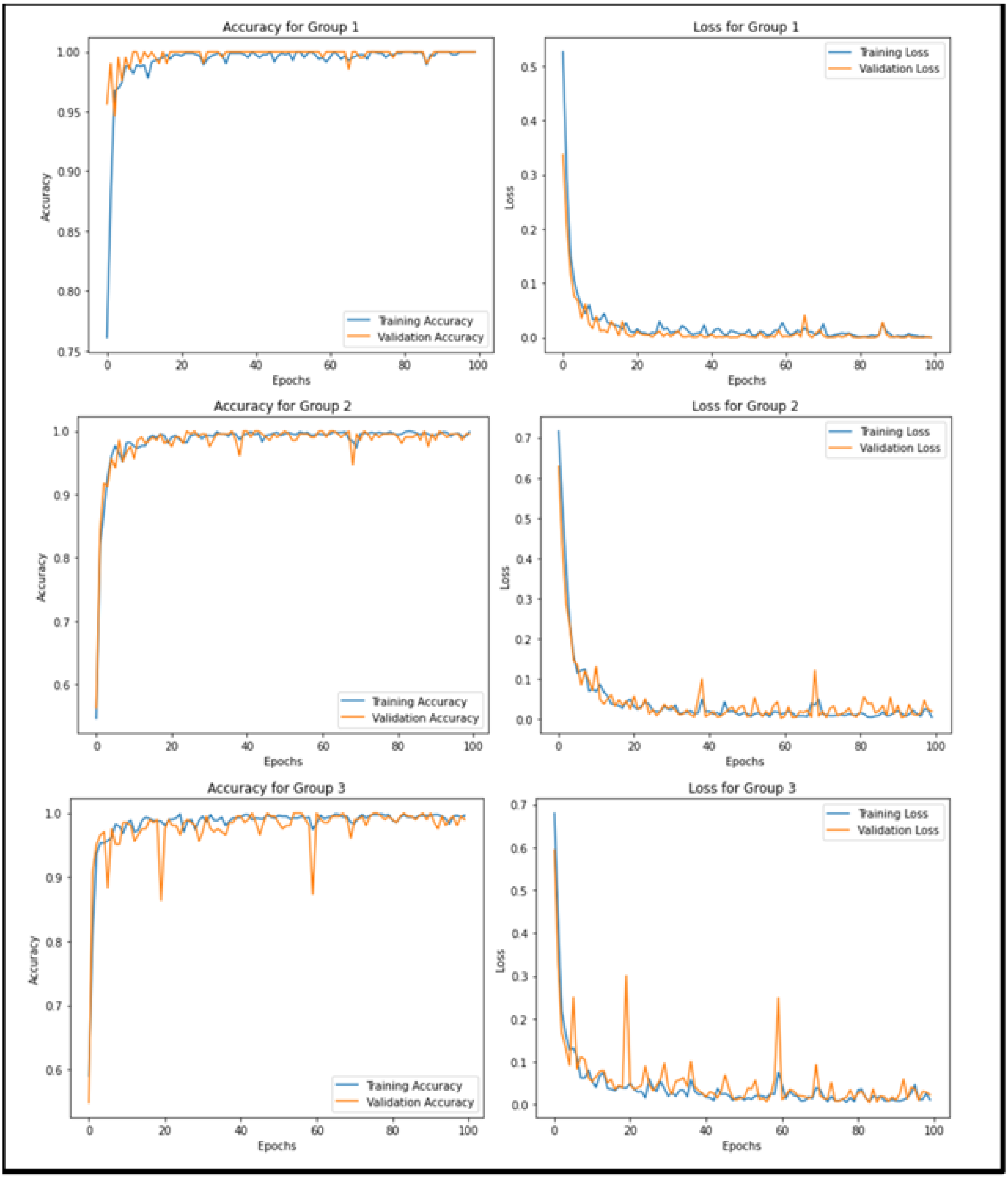


Figure 9: Prediction accuracy of the proposed Parallel CNN model.(a) Model accuracy and loss for Bacterial Spot vs. Healthy,(b) Model accuracy and loss for Early Blight vs. Healthy,(c) Model accuracy and loss for Late Blight vs. Healthy.

A l-Noor Journal for Information Technology and Cybersecurity, Vol.3, No.1, 2026 (27-47)

ISSN: 3078-5367 **DOI:** https://doi.org/10.69513/jncs.v3.i1.a4

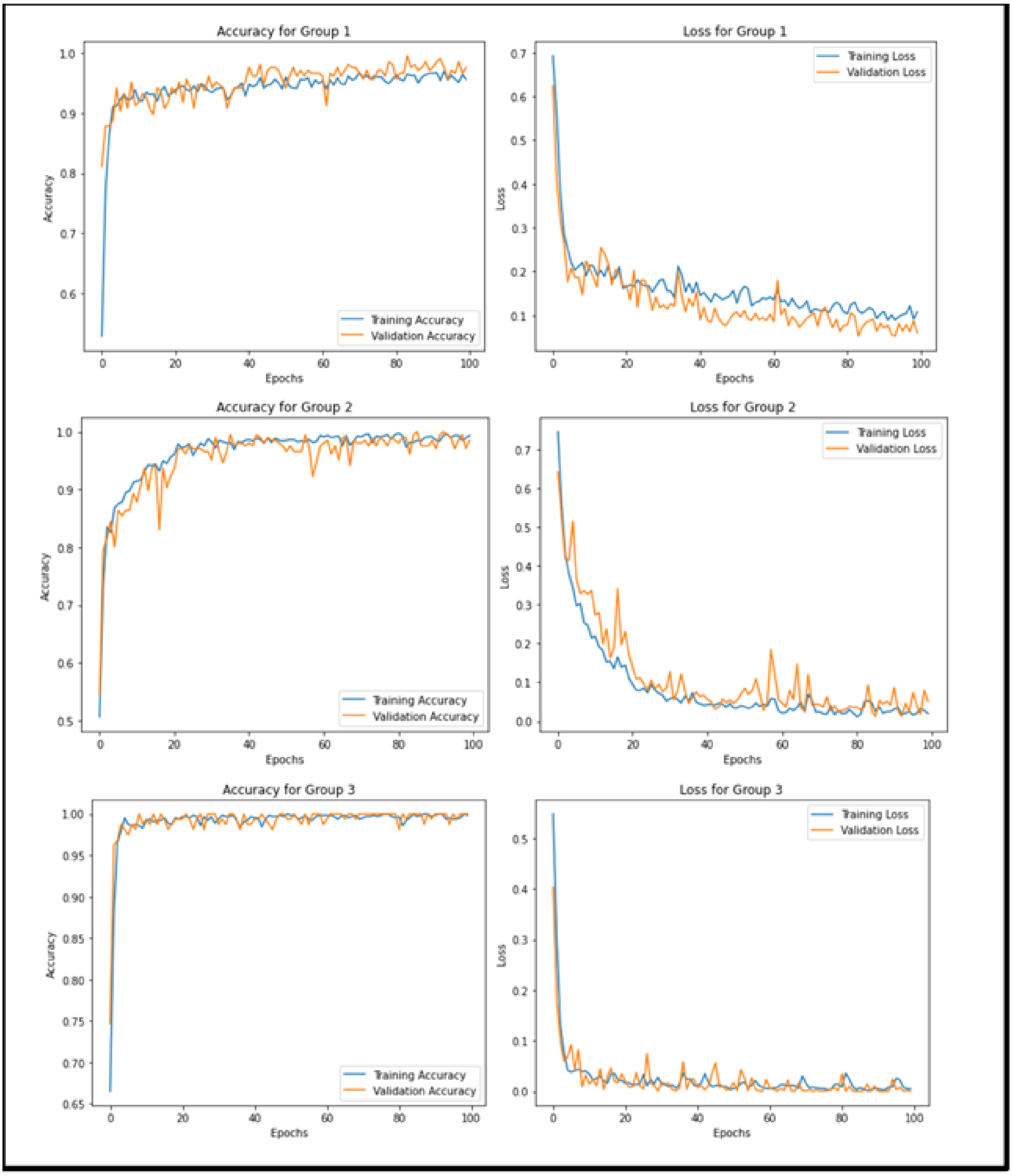


Figure 10 : Prediction accuracy of the proposed Parallel CNN model.(a) Model accuracy and loss for Apple Scab vs. Healthy,(b) Model accuracy and loss for Black Rot vs. Healthy,(c) Model accuracy and loss for Cedar Apple Rust vs. Healthy.

ISSN: 3078-5367 **DOI:** https://doi.org/10.69513/jncs.v3.i1.a4

Table 9: Classification Accuracy of the Proposed Parallel CNN Model for Different Plant Diseases

Plant	**Disease**	**Accuracy**
Corn	Cercospora Leaf Spot	98.54%
	Common Rust	100%
	Northern Leaf Blight	100%
Tomato	Bacterial Spot	100%
	Early Blight	100%
	Late Blight	98.54%
Apple	Apple Scab	95.63%
	Black Rot	100%
	Cedar Apple Rust	100%

The classification accuracy by the proposed Parallel CNN model, as revealed in Table 9, is high for all plant diseases tested. As far as the case with corn is concerned, high accuracy levels were registered by the classifier in all three disease classes, such that Cercospora Leaf Spot showed an accuracy level of 98.54%, while both Common Rust and Northern Leaf Blight demonstrated accuracies of 100%, which can mean that the model was able to successfully learn the features of diseases in question. In addition, similar high accuracy levels were reported for tomato diseases, as Bacterial Spot showed an accuracy of 100%, along with 100% accuracy in Early Blight disease and 98.54% accuracy in Late Blight disease, proving to be quite versatile with different diseases and different infections in general. In relation to apple, the accuracies were 100% for both Black Rot and Cedar Apple Rust diseases; however, Apple Scab demonstrated a lower accuracy of 95.63%, suggesting that the particular disease may have been difficult to identify due to symptom similarity. To further validate the efficiency of the suggested parallel architecture, an extra CNN architecture that is not parallel was assessed under similar experimental settings. The single CNN architecture attained a classification accuracy of 80.544%. This is considerably lower than the classification accuracy of the suggested parallel CNN architecture. Therefore, this finding suggests that there was no enhancement in the accuracy of the model due to the increment in its complexity (depth and width). Instead, this suggests that there is high efficiency in the suggested parallel feature extraction and fusion process.

Accuracy alone cannot determine the performance of the plant disease classification model, as it does not reflect the capability of the system in recognizing the different classes of diseases. Hence, other metrics such as Precision, Recall, and F1-Score have to be used in determining its performance.

Precision is the ratio of plant images correctly predicted by the model to a particular disease class out of all images that were predicted by the model belonging to that class. It is computed using Equation (.44):

$$\textbf{Precision} = \frac{\textbf{True Positive}}{\textbf{True Positive+False Positive}} \;\textbf{....(4.4)}$$

Recall measures the capability of the model to detect all actual cases of a given plant disease. It is the ratio of diseased samples classified correctly to all samples that actually belong to a given class of disease. Recall is defined in Equation (.45):

$$\textbf{Recall} = \frac{\textbf{True Positive}}{\textbf{True Positive+False Negative}}\textbf{.......(4.5)}$$

F1-Score A balanced measure can be obtained by combining Precision and Recall into one single value - the F1-score. In particular, the F1-score is useful when comparing datasets with class imbalance issues. It can be determined based on the formula shown in Equation (.46):

$$\textbf{F1 Score} = \textbf{2} * \frac{\textbf{Precision}*\textbf{ Recall}}{\textbf{Precision+ Recall}}\textbf{.......(4.6)}$$

The precision, recall, and F1-score are performance evaluation metrics used to assess the classification quality of the proposed system. Figure 11 illustrates these metrics for each plant disease class.

ISSN: 3078-5367 **DOI:** https://doi.org/10.69513/jncs.v3.i1.a4

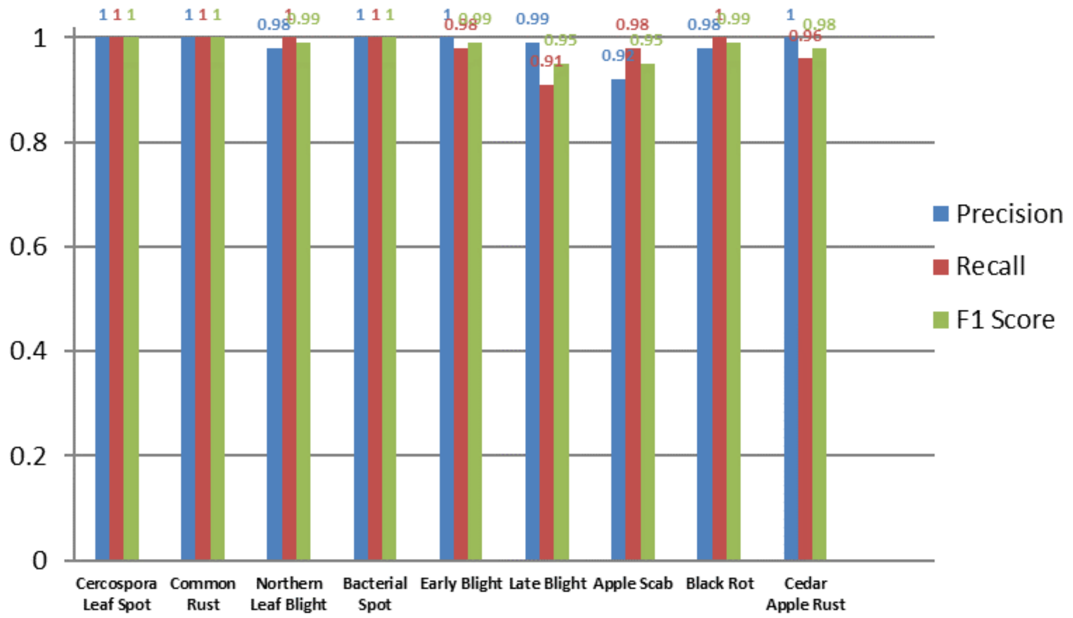


Figure 11 : Precision, Recall, and F1-Score for different plant diseases using the proposed Parallel CNN model

To give an idea of the performance of the model, Figure 12depicts the confusion matrix for each plant disease by classifying every disease independently against its respective healthy class. These include Healthy vs. Cercospora Leaf Spot, Healthy vs. Common Rust, and Healthy vs. Northern Leaf Blight for corn. Classification regarding tomato plants is done between Healthy vs. Bacterial Spot, Healthy vs. Early Blight, and Healthy vs. Late Blight. For apple, it classifies Healthy vs. Apple Scab, Healthy vs. Black Rot, and Healthy vs. Cedar Apple Rust. These confusion matrices present a clear pictorial view of how effectively each disease has been identified from its healthy state by the proposed Parallel CNN model.

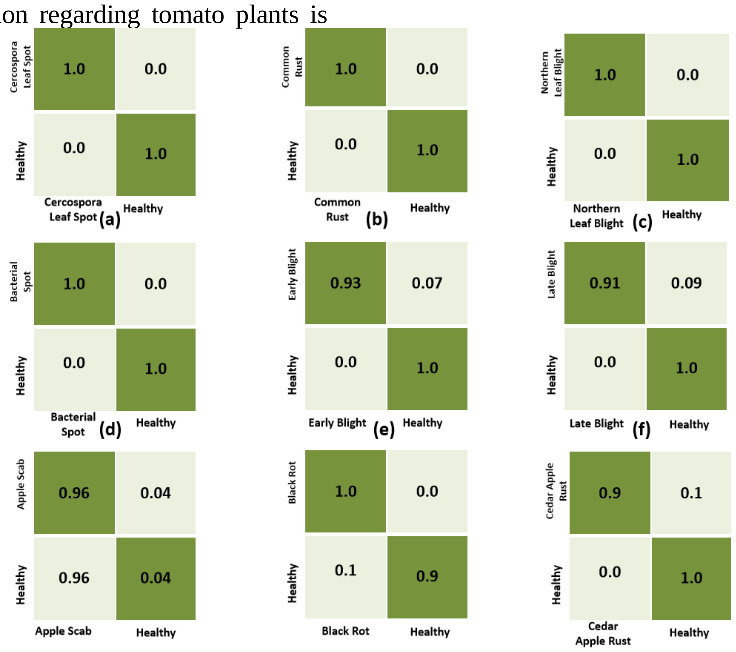


Figure 12: Confusion matrices of all plant diseases compared with Healthy samples across corn, tomato, and apple crops.

ISSN: 3078-5367 **DOI:** https://doi.org/10.69513/jncs.v3.i1.a4

The experimental evaluation of the system consisted of testing three independent binary models, each trained to distinguish a healthy leaf from one specific disease. Each model responded exclusively to the disease it was trained for. For instance, an image containing a specific disease resulted in only one model giving a positive output while the other two correctly classified the same image as healthy. These properties repeated well when parallel and sequential testing of the models was performed. Further verification of this property was done by processing the image of the first disease group through the second model, which correctly classified it as healthy. Thus, the above properties demonstrate selective activation and high fidelity classification in this parallel architecture. Overall, the results indicate that it is possible for the system to do a proper and efficient identification of multiple diseases without adding more model complexity, providing a fast and scalable solution that would be suitable for real-time agricultural applications.

In addition, the proposed convolutional neural network was tested on a carefully collected dataset containing images of corn, tomato, and apple leaves in both healthy and diseased conditions. These images have been extracted from nature to capture true variation in parameters like lighting, background complexity, orientation, and texture of the leaves. In general, the preprocessing of these images involved resizing and normalization in order for them to conform to the format required by the model input. Due to the limited number of collected real-field images and the difficulty of obtaining fully annotated data under natural agricultural conditions, this dataset was not used for training or quantitative evaluation, but only for external qualitative validation.The results show that the proposed model performs in such a way that all the healthy leaves are consistently identified as "Healthy" and the leaves with disease symptoms are correctly labeled as "Diseased." Confidence scores are uniformly high in the predictions, which again suggests that the model generalizes well beyond the controlled conditions of the PlantVillage dataset. However, these observations should be interpreted as qualitative indicators rather than statistically validated performance metrics.

The model thus appears quite practical for the detection of plant diseases under real-world conditions. It reliably distinguishes healthy and diseased leaves of various plants and therefore could be considered a robust and reliable model that has the potential to serve as a very useful tool for farmers and other agricultural workers who need rapid and accurate on-site assessments of plant health.

4.3 Comparison with a Lightweight Benchmark Network

In addition to the comparison with the VGG16 network, the proposed compact network was also compared with the MobileNet network, which is one of the prominent lightweight convolutional neural networks.

The proposed compact network was compared with the MobileNet network to evaluate the performance of the proposed lightweight parallel CNN with the state-of-the-art lightweight networks. Both networks were trained and tested with the same conditions, including the size of the input image (256x256x3), the number of classes (2-class classification), and the PlantVillage dataset.

As described in Table 4.4, the proposed lightweight parallel CNN achieved 99.19% accuracy, which is significantly higher than the 97.47% accuracy of the MobileNet network. The proposed lightweight parallel CNN also requires fewer trainable parameters, with 710,786 trainable parameters, whereas the MobileNet network requires 3,209,026 trainable parameters and 21,888 non-trainable parameters because of the Batch Normalization layers. The proposed lightweight parallel CNN also requires fewer computational operations, with 1.3272 GFLOPs, whereas the MobileNet network requires 6.3056 GFLOPs, which is 4.75x more than the operations required by the proposed lightweight parallel CNN.

From this comparison, it is evident that although the MobileNet is considered a lightweight network, it is still over-parameterized for this plant disease classification task. The proposed network not only reduces the size and computation of the network but also provides better classification accuracy, making it highly suitable for low-power, real-time agricultural applications.

In addition, more performance metrics have been considered for evaluating the efficiency of the two models. The latency in performing inference using the proposed model is 32.53 ms/image, while it is 35.13 ms/image for MobileNet. Furthermore, the memory footprint usage for the proposed model is 2.710479736328125 MB, while for MobileNet, it is 12.24146270751953 MB. Clearly, the proposed model is more efficient than MobileNet. Despite training both models for 100 epochs, the proposed model has exhibited more stable convergence during the training process.

In the end, the improved performance of the proposed network can be ascribed to the fact that it is a task-specific lightweight architectural design, which was carefully customized according to the characteristics of the PlantVillage dataset for the binary classification of plant diseases. Unlike the MobileNet architecture, which is a generic lightweight CNN architecture that is supposed to be generalizable across a broad spectrum of computer vision tasks, the proposed network is specifically designed for the extraction of only the most relevant spatial and texture features for the discrimination of healthy versus diseased leaves. The proposed network is able to achieve higher performance with significantly fewer parameters and reduced computational complexity.

ISSN: 3078-5367 **DOI:** https://doi.org/10.69513/jncs.v3.i1.a4

Table 10: Comparative evaluation of the proposed lightweight parallel CNN versus MobileNet

Model	Accuracy (%)	Trainable Parameters	Non-trainable Parameters	GFLOPs	Inference Latency (ms)	Memory Footprint (MB)	Training Setting
Proposed Lightweight CNN	99.19	710,786	0	1.3272	32.53	2.7104	100 epochs
MobileNet	97.47	3,209,026	21,888	6.3056	35.13	12.2414	100 epochs

4.5 Validation of High Accuracy and Generalization Analysis

In order to confirm the claimed high accuracy and to prevent any possible overfitting problem, the dataset was divided into 80% training and 20% testing parts based on stratification. Only training data were augmented (rotation, shifting, zooming, and flipping). In addition to the internal test accuracy, the model was tested on an external real-world agriculture field dataset, where the variation in illumination, occlusion, and background was observed. It proves the good generalization of the method beyond artificial datasets.The key to achieving the high accuracy is the specialized design of the proposed lightweight parallel CNN to detect the most discriminative spatial and texture information of the images for this task. Also, the application of parallel binary classifiers makes classification decisions less susceptible to the occurrence of error.Finally, the stable performance obtained after repeated training with fixed settings (the number of epochs equal to 100) demonstrates that there is no overfitting.

5. Conclusion

Based on the experimental results, the suggested lightweight CNN model can achieve very high classification accuracy despite having a low computational cost. It is because of the compact nature of the CNN model, which uses fewer layers and parameters than other existing models. Furthermore, due to its parallel structure, feature extraction in this model can be performed effectively and efficiently. In addition, the robustness of the lightweight CNN model can be seen from its performance in handling different plant leaves' real-world images, as such conditions are usually associated with changes in environmental factors. However, the architecture of the model follows a parallel binary decision paradigm, where each classifier performs only one function by recognizing just one disease. This may limit the ability of the model to diagnose two or more co-occurring diseases in one leaf sample. Moreover, a slightly lower accuracy level was seen in some of the classes especially in Apple Scab. The reason behind this phenomenon might be the similarity of symptoms with respect to other types of apples' leaf diseases, as well as the changes in lesions when exposed to different lightings. There are various directions to consider regarding the improvement of the proposed algorithm. First, the model can be assessed using hardware simulation platforms like gem5 to study execution time, architectural efficiency, and overall system-level performance in an ARM processor. Another direction includes using hardware accelerators in conjunction with optimization inference algorithms in order to minimize latency while maximizing throughput.In addition, using the proposed model for edge-AI hardware platforms such as Raspberry Pi and NVIDIA Jetson Nano may add practical value to the application. Lastly, scaling up the approach to cover different staple crops with massive amounts of data can improve generalization.

References

[1] Y.-Y. Zheng, J. Kong, X. Jin, X. Wang, and M. Zuo, "CropDeep: The crop vision dataset for deep-learning-based classification and detection in precision agriculture," Sensors, vol. 19, no. 5, p. 1058, 2019. https://doi.org/10.3390/s19051058

[2] F. Mohameth, C. Bingcai, and K. A. Sada, "Plant disease detection with deep learning and feature extraction using PlantVillage," Journal of Computer and Communications, vol. 8, no. 6, pp. 10–22, 2020. https://doi.org/10.4236/jcc.2020.86002

[3] K. Thenmozhi and U. S. Reddy, "Crop pest classification based on deep convolutional neural network and transfer learning," Computers and Electronics in Agriculture, vol. 164, p. 104906, 2019. https://doi.org/10.1016/j.compag.2019.104906

[4] F. Ahmad, M. Abdullah, H. Moon, and D. Han, "Plant disease detection in imbalanced datasets using efficient convolutional neural networks with stepwise transfer learning," IEEE Access, vol. 9, pp. 140565–140580, 2021. https://doi.org/10.1109/ACCESS.2021.3119000

[5] B. Wang, "Identification of crop diseases and insect pests based on deep learning," Scientific Programming, vol. 2022, Article ID 9179998, 2022. https://doi.org/10.1155/2022/9179998

ISSN: 3078-5367 **DOI:** https://doi.org/10.69513/jncs.v3.i1.a4

[6] Y. Sun, X. Yang, X. He, and X. Wu, "Northern maize leaf blight detection under complex field environment based on deep learning," IEEE Access, vol. 8, pp. 33679–33688, 2020. https://doi.org/10.1109/ACCESS.2020.2973658

[7] A. Fuentes, S. Yoon, and D. S. Park, "Deep learning-based techniques for plant disease recognition in real-field scenarios," in Advanced Concepts for Intelligent Vision Systems, Springer, 2020. https://doi.org/10.1007/978-3-030-42644-5

[8] Y. Sun and H. Huo, "Corn leaf disease recognition based on improved EfficientNet," IET Image Processing, vol. 19, no. 1, p. e13288, 2025. https://doi.org/10.1049/ipr2.13288

[9] A. J. Bleasdale and J. D. Whyatt, "Classifying early apple scab infections in multispectral imagery using CNNs," Artificial Intelligence in Agriculture, vol. 15, no. 1, pp. 39–51, 2025. https://doi.org/10.1016/j.aiia.2025.01.003

[10] S. G. Paul et al., "A real-time application-based convolutional neural network approach for tomato leaf disease classification," Array, vol. 19, p. 100313, 2023. https://doi.org/10.1016/j.array.2023.100313

[11] I. N. Yulita, N. A. Amri, and A. Hidayat, "Mobile application for tomato plant leaf disease detection using dense CNN," Computation, vol. 11, no. 2, p. 20, 2023. https://doi.org/10.3390/computation11020020

[12] M. Khan et al., "Deep transfer learning for fine-grained maize leaf disease classification," Journal of Agriculture and Food Research, vol. 16, p. 101148, 2024. https://doi.org/10.1016/j.jafr.2024.101148

[13] A. S. M. Siam et al., "SE-VGG16 MaizeNet: Maize disease classification using deep learning and squeeze-and-excitation attention networks," in Proc. IEEE ESCI, 2024, pp. 1–6. https://doi.org/10.1109/ESCI59607.2024.10497322

[14] K. Ahadian et al., "Maize disease classification using transfer learning and convolutional neural network with weighted loss," Heliyon, vol. 10, 2024. https://doi.org/10.1016/j.heliyon.2024.e30000

[15] Q. Wu, L. Xu, and E. D. Goodman, "Tomato leaf disease identification and detection based on deep convolutional neural network," Intelligent Automation & Soft Computing, vol. 28, no. 2, 2021. https://doi.org/10.32604/iasc.2021.018000

[16] Y. Jiang et al., "Real-time detection of apple leaf diseases using deep learning approach based on improved CNN," IEEE Access, vol. 7, pp. 59069–59080, 2019. https://doi.org/10.1109/ACCESS.2019.2914929

[17] 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. https://doi.org/10.3390/agriculture11070617

[18] C. Özden, "Apple leaf disease detection and classification based on transfer learning," Turkish Journal of Agriculture and Forestry, vol. 45, no. 6, pp. 775–783, 2021. https://doi.org/10.3906/tar-2103-45

[19] H. Assad et al., "Apple diseases: detection and classification using transfer learning," Quality Assurance and Safety of Crops & Foods, vol. 15, no. SP1, pp. 27–37, 2023. https://doi.org/10.1556/QA.2023.0002

[20] L. Deng et al., "Deep-learning-based rice disease and pest detection on mobile phone," Agronomy, vol. 13, no. 8, p. 2139, 2023. https://doi.org/10.3390/agronomy13082139

[21] P. Mallick et al., "Deep learning-based automated disease detection and pest classification in mung bean," Multimedia Tools and Applications, vol. 82, pp. 12017–12041, 2023. https://doi.org/10.1007/s11042-022-13245

[22] W. Chen et al., "Multispectral CNN for precision agriculture applications," Computers and Electronics in Agriculture, vol. 189, p. 106408, 2021. https://doi.org/10.1016/j.compag.2021.106408

[23] D. P. Hughes and M. Salathé, "An open access repository of images on plant health to enable the development of mobile disease diagnostics," arXiv preprint arXiv:1511.08060, 2015. https://doi.org/10.48550/arXiv.1511.08060
UNIVERSITY