Improving Leaf Disease Classification Using Dynamic Hyper Parameter Optimization of CNN Models

Bh. Prashanthi [1,2,*], A.V. Praveen Krishna [1], and Ch. Mallikarjuna Rao [2]

[1] Department of Computer Science and Engineering, Koneru Lakshmaiah Education Foundation, Guntur, India
[2] Department of Computer Science and Engineering, Gokaraju Rangaraju Institute of Engineering and Technology, Hyderabad, India
Email: bhupathi.prashanthi@gmail.com (B.P.); praveenkrishna@kluniversity.in (A.V.P.K.); professorcmrao@gmail.com (C.M.R.)
*Corresponding author

***Abstract*—Early leaf diseases are hard to identify based on crop type, environment, or symptoms. While there have been a number of deep learning algorithms applied to this topic, the vast majority of them are region-only and unable to differentiate between different kinds of crop types or diseases. To address this, this work employs an improved Convolutional Neural Network (CNN) model to recognize and classify plant leaf diseases. The improved CNN model uses leaf pictures and hyperparameters like learning rate and optimizer to identify illnesses. The technique was tested using the New Plant Disease Dataset. The study also showed how learning rate and optimizer affect model training. On the dataset for leaf disease, new optimized CNN model with enhanced reduced Learning Rate (LR) feature got the accuracy of 99.58% when the learning rate is 0.0001and got 95.59% accuracy where the learning rate is 0.0005 with a multi-class average 99.02%. The proposed model was also evaluated on New Plant Disease Dataset, other sequential CNN models, and transfer learning models and established higher accuracy and reduced the computational complexity. Precision from the model can be predicted as 97.53% and recall also as 97.53%, whereas, the F-measure can be predicted as 97.33%. According to the intrinsic analysis, it can be concluded that the impact of the optimizer option lies in the fact that the model reaches approximately the same performance level when it is set on the corresponding batch size and learning rate. Furthermore, it was found that lower learning rates provided a better performance than the higher learning rates indicating that using small batch size in conjunction with a small learning rate, produced the most effective model.**

***Keywords*—optimized Convolutional Neural Network (CNN) model, hyper parameters, reduced learning rate, optimizer, leaf disease classification**

I. INTRODUCTION

The agriculture industry has become one of the country's primary businesses and a major contributor to its GDP. Crop damage diminished production and hurt the economy.

Leaf limbs are diseased first because they are the most sensitive. Thus, the subject of how to address this issue demands research into crop management methods. Pray, monitor, and treat plant illnesses from conception until maturity before harvest. At various phases of cultivation, it is plagued by illnesses. It is thus equally advantageous to detect and control these diseases at the early stages to encourage high yield and product quality. However, because there are many thousands of acres of land under each farmer's stewardship and ten million varieties and sub-varieties of infectious diseases and because one leaf can be infested with several diseases, such identification is extremely impractical [1]. Knowledge of an expert in agriculture is absent in remotes areas and it is time-consuming.

Digital image processing and analysis enhance knowledge dissemination in biological sciences and agriculture [2]. An important type of agricultural automation involves plant leaves classification and growing healthy plants because the disease affects the parts of the plant leading to low crop production. There are capacitive impacts on the future development of the country, and the advancement of image processing technologies has once again awakened the interests of increasing crop yields. This is because over use of fertilizers and pesticides used in the fight against bacteria is also dangerous for crops. So, adopting appropriate methods is vital for farmers in order to grow good plants and experience maximum production.

While diagnosing diseases with the bare eyes some people do not see it or even if they see it they don't know the right cure. To diagnose plant diseases early and decrease agricultural damage, new plant categorization research methodologies must be developed. It has also established that the Deep Learning (DL) and Machine Learning (ML) algorithms are useful in identifying the health state of plant leaves from a disease perspective through image analysis [3]. Classification of leaves is

Manuscript received April 14, 2025; revised May 9, 2025; accepted June 18, 2025; published September 12, 2025.

doi: 10.12720/jait.16.9.1307-1317

important in the agriculture industry because it affects a country's economy indirectly. Before attempting to distinguish plants whose leaves are healthy and those with early sign of infection, a lot of precaution has to be observed. In the current generation, image processing is has come a long way that enables the classification of plants with little or no human interaction.

Deep learning algorithms have been widely used in several industries because to their effectiveness in pattern recognition and autonomous driving. Against this background, this research proposal aims at determining the use of deep learning approaches in the cultivation environment to enhance on crop management since the demand for food production is rising rapidly [4]. Thus, the agricultural industry has turned its attention to deep learning solutions. Meaningful results have now been achieved by deep learning algorithms in a range of operations, including crop and weed identification, fruit picking, and leaf characterization [5]. Through meta-learning, deep learning has simplified product identification and categorization. With its convolutional neural network image processing, deep learning is a strong discovery tool.

The main focus of the work is the identification and comparison of the diseases that affect the leaves, with a view of aiding large farming industries in cheap detection of the diseases thereby enhancing production of crops. This project aims to improve the Convolutional Neural Network (CNN) model to categorize leaf diseases. The study advises using dynamic hyperparameters to increase model rate. The given research work recommends that the setting of dynamic hyperparameters like the learning rate and the regulazers can progress the accurateness of the model particularly as applied to different data. The research also presents a metric for measuring the performance of the models and evaluating between one model and another. The convenes of the study also indicate that the proposed CNN model with adaptive hyperparameters of dynamic nature yields more accurate classification than the normal CNN models.

II. Related Work

Plant diseases cause major losses on leaves in plants and crops and hence a reduction in both the quantity and quality of the produce. Diseases develop on the foliage through a range of areas including fungus, bacterias, viruses and weather conditions, such as moisture or drought. General information about the symptoms of leaf diseases reveals that a wide range of signs can be associated interchangeably with different types of diseases and plant varieties. There are several signs that usually indicate the presence of diseases on leaves and some of them are as follows: Often the leaves develop wrong colors, possibly they spot, they wither in appearance, they may develop an improper shape or size, and lastly, they may fall off the plant [6]. In some cases, all parts of the plant may be infected and stunted, yield reduced or killed outright.

Diagnostic and disease categorization are important in the prevention of the diseases. This can be effected by use of techniques like, visual, lab and remote sensing. Machine learning has been used to automate plant leaf disease diagnosis and categorization based on collected images. In this way it may be possible to refine the accuracy and efficiency by which certain diseases are detected and in doing so, might be able to reduce the fiscal impact leaf diseases have on agriculture.

A. Types of Leaf Diseases

Several forms of leaf diseases can be present in the plant and crops [7]. Other classes of diseases that affect the leaves include

- Fungal diseases: These are disease agents that invade plant organs causing different signs such as spots, blights, rusts and mildews.
- Bacterial diseases: These are caused by bacterial pathogens, which invade tissues of plants and bring about wilting, leaf spot, blight and other disorders.
- Viral diseases: These are those which are caused by viral pathogens that can enter tissues of plant and lead to disease symptoms such as mottle, yellowing and stunt.
- Nematode diseases: These are diseases that are as a result of nematodes that prefer to feed from the roots or leaves of the plants and they exhibit signs such as yellowing, wilting, and stunted growth.
- Abiotic diseases: These are due to environmental factors such as temperature, humidity and soil types and the signs includes yellowing of leaves, wilting and leaf curling.
- Insect and mite damage: This results from several insect and mite species that feed on parts of the plants and may lead to situations such as twisted leaf, different colored, or holed.

There are several generic types of diseases affecting the leaves of the plant, they are: powdery mildew, rust, anthracnose, blight, and leaf spot. The kind and intensity of the disease in relation to the foliage may differ with the species of the plant, geographical location and methods of its usage.

B. Comparison of CNN Models for the Identification and Classification of the Leaf Disease

One of the most investigated fields of machine vision, agriculture has led the way in using computer vision-based systems to diagnose crop diseases. Conventional methods often make use of prior established algorithms, features and classifiers, however with an exploration of the CNNs, image detection can be implemented to detect near-anomalies on the plants natural surroundings. However, the identification met with several obstacles in a complex ecosystem in nature, these traditional methods face challenges due to issues such as poor contrast, the presence of lesions with a different size and type, have small differences. Machine learning will help farmers, thereby increasing profit and making the agricultural industry more sustainable.

In Ref. [8], the authors introduced image processing techniques such capture, preprocessing, feature extraction, and classification. They also present many image processing plant disease identification case studies. The paper suggests that image processing can boost crop productivity and reduce agricultural losses.

In Ref. [9], a technique of image feature analysis for automating the recognition of leaf diseases in a number of crop species is outlined. The authors have invented a system to diagnose the maladies affecting the foliage using both machine learning and image analysis. They also present a number of experiments that have been carried out on different crop species and showed their method can be used to recognize some of the common leaf diseases. In aggregate, the article indicates that the approach to the automated detection of the diseases on the leaves can be effective for increasing the yields and decreasing the losses in the agriculture.

Deep learning is used to identify plant diseases and parasites in Ref. [10]. Convolutional Neural Networks for plant disease and parasite identification are the authors' last topic. Third and final, they demonstrate deep learning's ability to recognize several plant illnesses and parasites. Thus, the article hypothesizes that deep learning can identify pests, saving the agriculture business money. Additionally, deep learning approaches may be used in complex future applications like precision agriculture and crop management [10].

As Ref. [11] noted, CNNs are widely employed in plant diseases. Plant disease diagnosis and early detection are discussed by the writers. Additionally, they describe picture classification CNNs and plant disease detection CNN models. Experiments show how the CNN model can identify plant diseases including tomato leaf blight, potato late blight, and grape leaf blight. CNN-based ways to diagnose plant illnesses let farmers to address them early, according to the authors. Convolutional neural networks may also be used in precision agriculture and agricultural yield prediction [11].

In general, it is believed that hazardous pests can negatively affect both the methods of manufacturing and companies' products in many industries [12]. Thus, being able to identify pests is important in these production processes in order that management decisions concerning control of the pests may be made. The challenge however, is that most insect species are nearly indistinguishable and many of their attributes are nearly identicals especially when speaking with people who have dealt with traps before. This research proposes an approach for classifying various types of caught insects that employs deep CNN modeling and currently existing images to make predictions. In the evaluation, a database of 200 images of a candy factory containing about 3000 insects of six species was used. Consequently, the results showed that the average accuracy rate for detection was approximately 84% while that of classification was about 86% [12].

In a separate research paper, the author devised a ground-breaking deep learning method proposed in four phases comprising the suggested methodology [13]. The first pre-processing step is image enhancement, and the second is segmentation, whereas feature extraction deploy a trained CNN. This architecture is trained in the model training phase and the CNN is fine tuned in this process. In the classification step, Support Vector Machine (SVM) is used to classify the disease. Many researchers have employed significant various methodologies and designs in an attempt to study the CNN for this particular task. Many papers have described ways in which transfer learning and ensemble learning can be implemented to enhance the performance of CNN. Transfer learning has been used severally to train CNNs with little supervision since the features learned by pre-trained models can be used as base to enhance the model.

Recent advancements in plant disease classification have seen the emergence of Vision Transformers (ViTs) and Self-Supervised Learning (SSL) approaches as promising alternatives to conventional CNN-based methods. Vision transformers, originally introduced for natural image processing, have demonstrated strong capabilities in capturing long-range dependencies and complex spatial patterns, making them suitable for fine-grained image classification tasks. For example, Dosovitskiy *et al.* [14] proposed the ViT, which achieved state-of-the-art performance on various image benchmarks by dividing images into patches and applying transformer encoders without convolutional operations. In the agricultural domain, ViT-based models have shown high accuracy in identifying plant diseases while requiring fewer training samples when combined with data augmentation techniques.

Parallelly, self-supervised learning techniques such as contrastive learning and masked image modeling have enabled models to learn robust representations without the need for labeled data. Grill *et al.*'s [15] Bootstrap Your Own Latent (BYOL) and He *et al.*'s [16] Momentum Contrast (MoCo) are notable SSL frameworks that have significantly improved downstream classification performance.

In contrast to these transformer-based and self-supervised paradigms, the present study focuses on a carefully optimized CNN architecture with dynamic hyperparameter tuning, designed to balance classification accuracy and computational efficiency. While ViTs and SSL are effective, they often require extensive pretraining or large-scale hardware. This research positions itself as a lightweight yet accurate alternative, particularly suitable for real-time and resource-constrained agricultural environments.

Based on the literature study, there is a need to optimize the detection technique which can improve accuracy and effectiveness. Additionally, the enhanced CNN model can facilitate leaf disease categorization in dynamic environments. Large datasets of photos of diverse plant species and diseases can be used to train optimized CNN models.

III. METHODOLOGY

Several challenges persist in the existing literature on deep learning systems for plant disease detection, particularly the inadequate consideration of environmental factors, variations, and the diverse nature of leaf illnesses.

Early identification and treatment of these diseases are critical but remain difficult, especially in rural areas where access to agricultural experts is limited and time-consuming. Advances in image-based recognition have been made by minimizing reliance on extensive image pre-processing and leveraging CNNs' inherent ability to perform effective feature extraction [4]. However, the scarcity of large, diverse datasets for training such models continues to be a significant hurdle. The overall design process is summarized in Fig. 1 (procedure flowchart) and Fig. 2 (optimized CNN architecture).

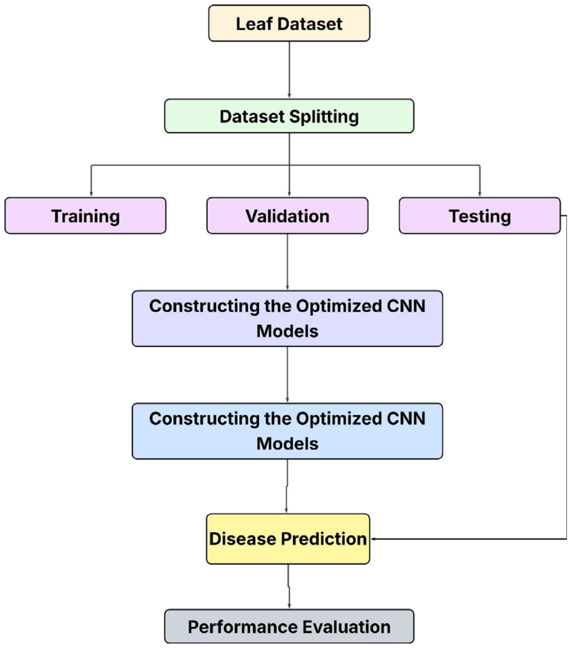


Fig. 1. Proposed methodology.

The primary contribution lies in the development of a customized CNN architecture that integrates three Conv2D blocks with an Inception layer, specifically designed for efficient multi-scale feature extraction in leaf disease classification. This architectural innovation enables the model to capture both fine and coarse-grained patterns essential for distinguishing visually similar disease classes. Additionally, the approach incorporates dynamic hyperparameter tuning—particularly adaptive learning rate scheduling and optimizer selection—which significantly enhances training efficiency and classification accuracy. Compared to existing approaches, which often rely on static architectures and fixed hyperparameters, this method demonstrates superior accuracy (99.58%) while maintaining computational efficiency with a reduced input resolution. This makes the model well-suited for real-world agricultural applications where resource constraints and rapid inference are critical.

The proposed CNN model introduces a novel architectural design tailored for multi-class leaf disease classification. It comprises 15 layers, including three sequential Conv2D blocks, each followed by batch normalization, ReLU activation, and max-pooling, allowing for progressive spatial feature extraction and dimensionality reduction. A key innovation is the integration of an Inception module, which employs parallel convolutional operations as shown in algorithm. This enables the network to capture both fine-grained and large-scale features simultaneously, enhancing its ability to distinguish between visually similar disease patterns. The model also incorporates dropout regularization (rate = 0.4) to mitigate overfitting and includes a dense fully connected layer for final classification.

Convolutional neural network has input, hidden, and output layers. Fig. 2 shows convolution, normalizing, pooling, and completely linked layers between the input and output layers [3]. This layer is called convolutional because it utilizes filters to extract and generate classification feature maps. It uses ReLU activation functions [10]. To comprehend CNN, consider these fundamental stages:

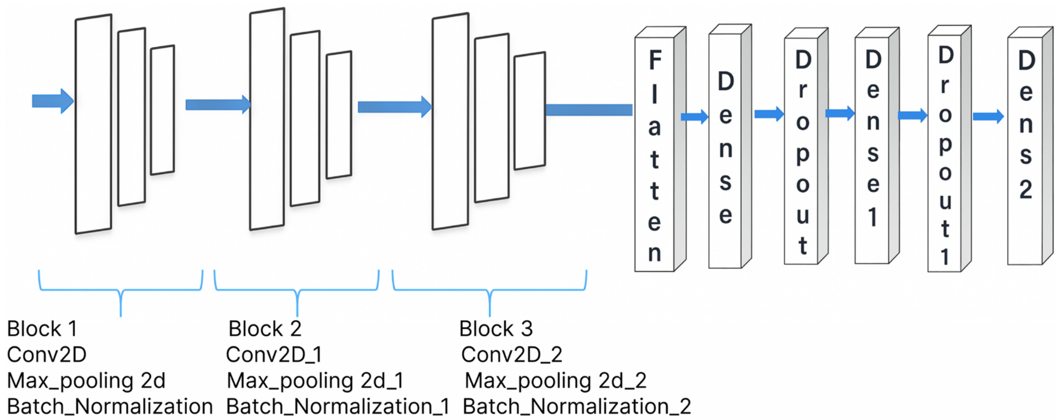


Fig. 2. Layered architecture description of the proposed optimized CNN model.

The input layer processes raw image data resized to dimensions of 32×32 pixels with a depth of 3 color channels. In the convolutional layers, filters perform dot products with image patches to generate feature maps; for instance, employing 12 filters produces an output volume of 32×32×12. Each convolutional output undergoes element-wise activation using functions such as ReLU, Tanh, or Sigmoid to introduce non-linearity. Pooling layers, typically max or average pooling, follow to reduce spatial dimensions, speed up computation, save memory, and prevent overfitting. Addressing the need for precise feature learning in leaf disease classification, this research introduces a novel, robust CNN model optimized through dynamic hyperparameter tuning, including adaptive learning rates and optimizer selection. Preprocessing techniques reduce image size to balance computational efficiency and accuracy.

A. Description of the Model

A simple stack of layers as shown in Fig. 2 with exactly one input tensor and one output tensor per layer is suitable for an optimized model. Many of the fundamental Keras algorithms are also found in the sequential class, where the input is fed and output is anticipated with trained and inferred results according to the needs. The data flow from top to bottom helps to make the layers more improved and informative, which are needed for modification and filtration [13]. The process of the developed optimized CNN algorithm for identification and classification of the diseases affected leaves is provided in Algorithm 1.

Algorithm 1: Optimized CNN Model for Leaf Disease Classification

```
Input: Leaf image dataset D
Output: Predicted disease classes and performance metrics
1: Preprocess Dataset D
      ‣ Resize images to 62 × 62 × 3
      ‣ Normalize pixel values
      ‣ Apply augmentation (flip, rotate, zoom)
2: Split D into training set Dtrain, validation set Dval, and test set Dtest
3: Initialize CNN Model
      ‣ Input Layer ← (62 × 62 × 3)
      ‣ for i = 1 to 3 do
          Add Conv2D Layer
          Add Batch Normalization
          Add ReLU Activation
          Add MaxPooling
       end for
      ‣ Add Inception Module (filters: 1 × 1, 3 × 3, 5 × 5 in parallel)
      ‣ Flatten Layer
      ‣ Add Dense Layer with Dropout (rate = 0.4)
      ‣ Output Layer ← Softmax Activation
4: Set Dynamic Hyperparameters
      ‣ Choose Optimizer ∈ {Adam, SGD, RMSProp}
      ‣ Initialize Learning Rate LR
      ‣ Define LR Scheduler: reduce LR after 5 epochs
5: Train CNN Model
     for epoch = 1 to MaxEpochs do
        Train on Dtrain with current LR and optimizer
        Validate on Dval
        if no improvement for N epochs then
           Reduce LR
        end if
     end for
6: Evaluate on Dtest
      ‣ Predict classes for test samples
      ‣ Compute Accuracy, Precision, Recall, and F1-Score
7: Return predictions and performance metrics
```

The feature extraction technique leverages a hybrid CNN architecture combining traditional Conv2D layers with an Inception module to effectively capture multi-scale spatial features. This design enables simultaneous extraction of fine local details and broader contextual information, which is critical for accurately distinguishing between visually similar leaf diseases. Unlike hand-crafted feature methods or single-scale CNNs, this approach automates hierarchical feature learning directly from raw images, reducing reliance on manual intervention and improving adaptability. Furthermore, the integration of batch normalization and max-pooling enhances feature robustness and computational efficiency, making this technique well-suited for large-scale, multi-class classification tasks in plant disease detection.

The optimized CNN model presented in this study features a modified layered architecture specifically designed for accurate identification and classification of leaf diseases [17]. This architectural adjustment was motivated by the complex nature of leaf image patterns, requiring enhanced capacity to capture intricate details. The model includes a sequence of layers dedicated to feature extraction, followed by classification layers. It incorporates three Conv2D blocks, including one Inception module, along with max-pooling and batch normalization to improve feature representation and stability. The complete architecture consists of fifteen layers, carefully arranged and fine-tuned to maximize prediction accuracy and robustness. Furthermore, dynamic hyperparameter optimization—particularly varying learning rates and optimizer types—was employed to investigate their impact on model performance, resulting in significant improvements in disease prediction and categorization.

B. Training, Testing, and Validation

The dataset is split into training, validation, and testing sets with ratios of 80%, 10%, and 10%, respectively. Model training employs backpropagation with the Adam optimizer. Dynamic hyperparameter tuning is applied, including a learning rate scheduler that reduces the learning rate after a fixed number of epochs to improve convergence and accuracy [18].

C. Hyper-Parameters in Deep Learning Model

Even with CNN research's success, finding a good model takes years of expertise, knowledge, and network training. Training hyperparameters determine network behavior [19]. Ideal model hyperparameters are difficult to define. To find the optimum hyperparameters, manual testing and algorithm tweaking on multiple settings are available now. Grid and random search tuning methods are laborious [19]. CNN networks have parameters and hyperparameters. The model's data processing changes the parameters.

Static hyper-parameters are established before model training and do not change. Some hyper-parameters include epochs, learning rates, and optimizer types. Learning rate determines how quickly the network updates optimizer parameters. Epochs are full runs across the data points. The batch size is samples processed each epoch. The optimizer will assign model-appropriate optimization techniques to improve accuracy and efficiency. Optimization methods include stochastic gradient descent and the Adam algorithm [19].

D. Evaluation Metrics

The models to be developed in this research work shall be assessed using the following relevant assessment matrices presented in this section. The changes were introduced to the set of images and they became the images of the augmented training set. It was also horizontal flipped, the range of rotation was 90° and the range of zoom was kept 0.2. The images were also rescaled before passing to the model.

1) Accuracy

Accuracy is "the ratio of the number of correct predictions made by a model to the total number of predictions" [20]. The formula for accuracy is given as:

$$Accuracy = \frac{Total\ Number\ of\ Correct\ Predictions}{Total\ Predictions}$$

2) Precision

Precision describes the portion of positive predictions, which was accurate [20]. The formula of precision is

$$Precision = \frac{Treu\ Positive}{True\ Positive + False\ Positive}$$

3) Recall

Recall stands for the proportion that draws effective actual positives [20].

$$Recall = \frac{Treu\ Positive}{True\ Positive + False\ Negative}$$

4) F1-Score

It demonstrates how many relevant documents are retrieved, and how many of the returned documents are relevant [20]. The formula of the F1-Score is as follows:

$$F1 - Score = \frac{2 \times (Precision \times Recall)}{Precision + Recall}$$

IV. Results and Analysis

Python libraries, TensorFlow [21], and Keras are used to test the optimized CNN model. The Adam optimizer, with a learning rate and loss function, was utilized for training. The accuracy or loss of the optimized CNN learning models developed on the New Plant Disease Dataset is studied regarding the count of training epochs. Table I shows the characteristic settings for plant disease detection models employing transfer learning.

TABLE I. Learning-Model-Training Parameters

S.No	Parameter used	Value
1	Training Epochs	10
2	Early stopping	Yes
3	Learning Rate	0.001, 0.0001, and 0.000,01
4	Batch Size	32
5	Drop out	0.4
6	Image Depth	3
7	Optimizer	Adam, SGD and RMSProp
8	Image size	62×62
9	Reduce LR and callback	Yes

A. Dataset Description

The dataset used in this study is the New Plant Disease Dataset [22], which consists of 54,306 images covering 38 distinct categories that include both healthy and diseased leaf samples across 14 different plant species such as tomato, potato, grape, and apple. To ensure balanced learning, the dataset was split into 80% training, 10% validation, and 10% testing sets. While minor class imbalances existed, their impact was mitigated through a series of preprocessing and augmentation techniques. These included horizontal flipping, rotation (up to 90°), zooming (range = 0.2), and rescaling, which collectively enhanced the diversity of training samples and helped in balancing underrepresented categories. Additionally, all images were resized to a uniform dimension of 256 × 256 during preprocessing to standardize input to the CNN model. This ensured that the model learned invariant features across varying lighting conditions, orientations, and disease severities, contributing to its high generalization capability across all 38 categories.

B. Pre-Processing the Data

Learn how to convert image for CNN. Since tensors contain knowledge rather than data, they are like multidimensional arrays. Before loading the dataset, the programs must include all the necessary imported libraries. Every process of data analysis, must incorporate this phase. Images, therefore, can come in different sizes and shapes in the process of designing. In data pre-processing process is standardizing the images. Seven different types of enhancements are used for the aggregated photos, where the classification is performed using the Image Data Generator tool and keeping an image size of 256 by 256. Fig. 3 depict the pre-Processed image with a dimension of 256×256.

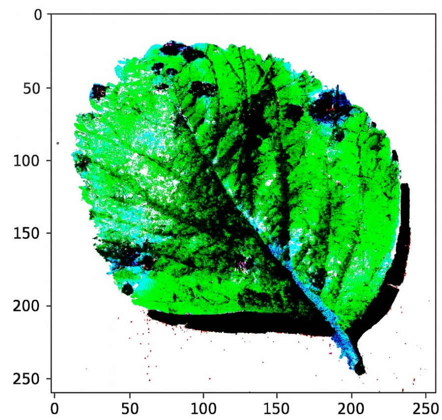
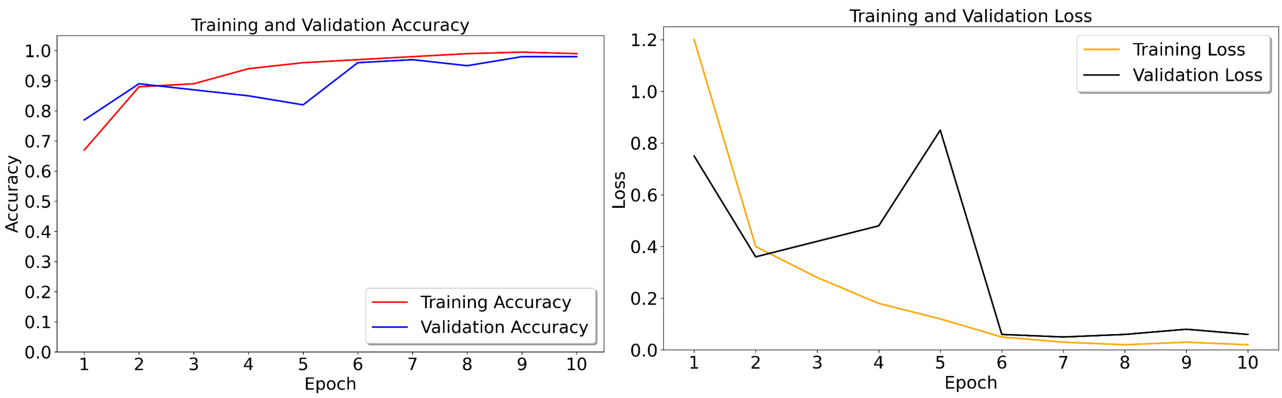


Fig. 3. Pre-processed image.

C. Proposed Optimized CNN Model Dataset Performance

Multiple distinct methods with early halting were used to train the plant disease model. The suggested method uses the CNN algorithm to categorize the bacterial and healthy leaves of leaves. CNN's accuracy rating is 99.02%. CNN is the model that best distinguishes between healthy leaves and bacterial leaves. Fig. 4 shows the optimized CNN model's performance in this study. Fig. 4

shows how the lowered learning rate and dynamic drift change performance accuracy and loss. This enhances the model's accuracy and performance over competitors. The achieved average multi-class accuracy is 99.02% on 38 classes of NewPlant Disease Dataset.



(a) (b)

Fig. 4. The fine-tuned CNN model performance on new plant disease dataset.

The optimization algorithm employs a dynamic learning rate scheduling mechanism to achieve a principled balance between exploration and exploitation phases. In the early training epochs, a relatively higher learning rate facilitates extensive parameter space exploration, enabling the model to escape shallow local minima and traverse diverse regions of the loss landscape. After a predefined epoch threshold, the learning rate is systematically decayed using a scheduler, thereby entering the exploitation phase. This controlled reduction allows the optimizer to perform fine-grained updates in the vicinity of promising minima, enhancing convergence stability and model generalization. The combination of adaptive learning rate drift with empirically tuned optimizers (Adam, SGD, RMSProp) ensures a robust trade-off between search diversity and convergence precision.

Table II presents a detailed performance comparison of the proposed optimized CNN model against several established architectures. The optimized CNN model with a reduced input size of 62×62 achieves the highest accuracy of 99.58%, surpassing deeper and more complex models such as ResNet9 [27] (99.19%), AlexNet [23] (98.74%), and MobileNetV3 [25] (96.41%). Notably, it significantly outperforms VGG-16 [24] (87.99%) and InceptionV3 [26] (91.84%), indicating the efficiency of the proposed lightweight architecture. The comparison also includes CNN variants trained on different input resolutions—224×224 [28] (95.33%), 210×210 (91.09%), and 256×256 (98.27%)—all of which show lower accuracy than the proposed model. These results emphasize the technical strength of the model's design, which combines efficient feature extraction and hyperparameter tuning to achieve superior classification performance with lower computational overhead.

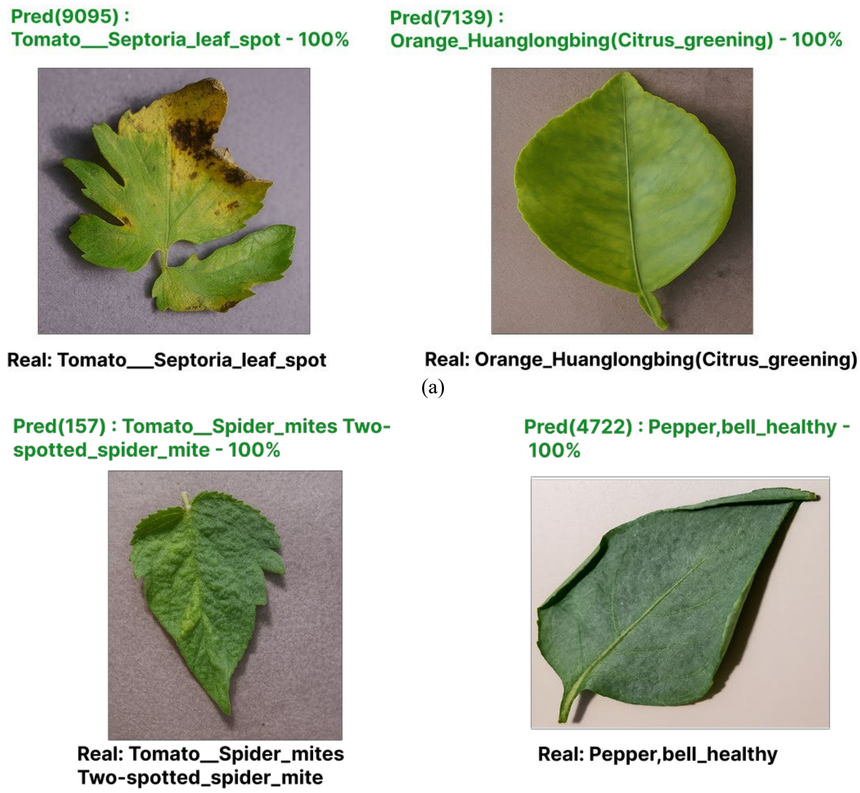
A comprehensive comparison has been performed against established models such as AlexNet, VGG-16, MobileNetV3, InceptionV3, and ResNet9, demonstrating the superior performance of the proposed CNN model. With an accuracy of 99.58%, the optimized model outperforms all other architectures across the same dataset, validating its effectiveness in multi-class leaf disease classification. These results confirm the model's competitive advantage in both accuracy and computational efficiency.

TABLE II. PERFORMANCE COMPARISON WITH OTHER CNN MODELS

Model	Accuracy (%)
AlexNet [23]	98.74
VGG-16 [24]	87.99
MobileNetV3 [25]	96.41
InceptionV3 [26]	91.84
ResNet9 [27]	99.19
CNN model with size 224×224 [28]	95.33
CNN model with size 210×210	91.09
CNN model with size 256×256	98.27
optimized CNN model with 62×62 (proposed)	99.58

D. Model Performance on Correct Prediction

The optimised CNN model's classification of healthy and damaged leaves on test images is evaluated. In this case, an image which is labeled correctly is one in which a model correctly describes an image of a healthy or diseased plant foliage. The assessment of the extent to which correctly predicted images mirror the intended design is ordinarily measured by accuracy. These measures indicate how accurate the model is working in distinguishing the images in line with their tags. These results reveal that, if the model scored a 90 percent accuracy level it means 90 percent of the images were labeled correctly by the model. The performance of the correct prediction of images is crucial since it displays the efficiency and recognition of healthy or diseased in a real environment. Further, it helps in determining sectors where the model might be performing optimally, and sectors that might need development. Fig. 5 illustrates shows that a fine-tuned CNN model can be used effectively to forecast images.



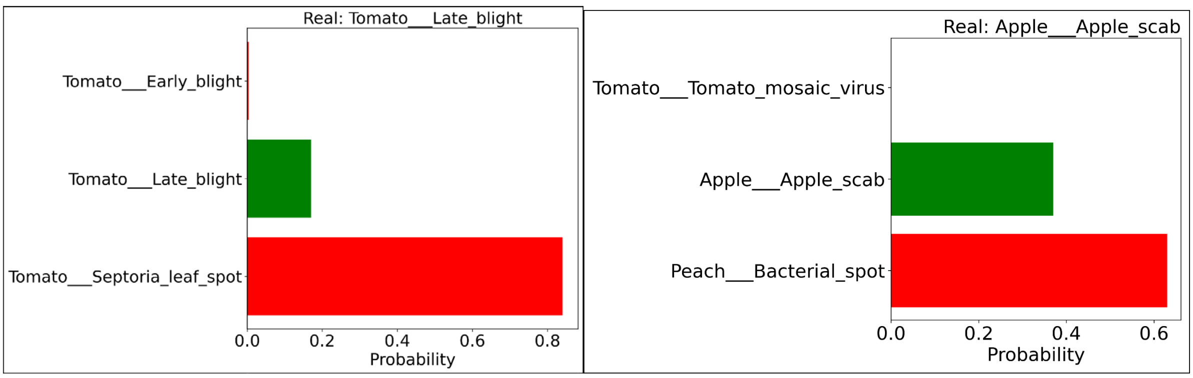
(a)

(b)

Fig. 5. The performance of the optimized CNN model on correctly predicted images.

E. Model Performance on Incorrect Predictions

The feature distribution on the images that were falsely predicted may help understand more about model inefficacies and areas of problematic enhancement. These are the images that we incorrectly predicted as the wrong class label according to the model. From these images, the patterns or features not discerned by the model that resulted in a wrong prediction are clear. The thick lines formally mean that the corresponding edges are abstracted from some node pairs because these node pairs do not have sufficient representing capability, which can help identify missed patterns or features when constructing the model architecture or tuning hyperparameters. Fig. 6 shows performance of improved CNN model on falsely predicted images.



(a) (b)

Fig. 6. Performance of improved CNN model on falsely predicted images.

F. Model Predictions on Test Data

Predictions on test data entail applying the learned CNN model to information that has never been seen before. The test data is diverse from the training dataset and is used to see how well the CNN is generalizing to unseen data.

Measures of accuracy are decided based on comparisons of defined class labels from both a training and test data set. These measure refer to how the model is faring well on the test data. This makes it possible to analyze the specific model's accuracy; misclassifications occurrences, and existing trends in the data set. Table III shows how well the enhanced CNN model predicts test data.

TABLE III. OPTIMIZED CNN MODEL PREDICTION PERFORMANCE ON TEST DATA

id	Apple__Apple_scab	Apple__Black_rot	Apple__Cedar_apple_rust	Apple__healthy	Blueberry__healthy	Cherry_(including_sour)__Powdery_mildew
TomatoEarlyBlight6	0.000121591	2.18728e−05	0.062278	0.00133807	3.32871e−09	0.000968041
TomatoYellowCurlVirus	2.76534e−14	1.18166e−20	3.85028e−14	1.20147e−12	3.27036e−14	6.67822e−11
TomatoYellowCurlVirus6	1.87469e−36	0	6.31038e−33	5.36285e−30	5.32808e−35	4.23579e−27
PotatoHealthy	2.25105e−12	4.80548e−13	4.06476e−12	1.05776e−06	2.62508e−07	6.05491e−09
TomatoYellowCurlVirus7	4.65022e−35	0	6.78762e−31	6.49932e−33	3.88735e−33	1.39707e−26

G. *Effect of Hyperparameters on Optimized CNN Models*

Thus, the proposed model algorithm's accuracy is determined for each hyperparameter during the experiment and Table IV defining the optimal accuracy of CNN model depending on hyperparameters. Following is the test precision for the optimized CNN being evaluated on the portion of the dataset that was not used for training. In order to demonstrate how each hyperparameter may be fine-tuned to achieve optimal performance, we also provide many graphs for these parameters.

TABLE IV. OPTIMIZED CNN MODEL PERFORMANCE FOR VARIOUS OPTIMIZERS AND LEARNING RATES

Optimized CNN Model	Optimizer	Learning rate	Accuracy
CNN Model 62 × 62	Adam	0.001	96.45
		0.0001	99.02
		0.00001	93.83
	SGD	0.001	91.23
		0.0001	98.01
		0.00001	89.8
	RMSProp	0.001	--
		0.0001	98.1
		0.00001	80.56

Table IV shows how learning rates affect optimizer performance. All Table IV examples utilized 32-batch sizes. As seen in Table IV, optimizers perform poorly when the learning rate is excessively high or low. Table IV accuracy "--" represents model is not trained. The optimizer is moving slowly toward the loss function when the learning rate is too high, generating a local or global minimum. The optimizer becomes unstable and fails or rapidly converges to a local minimum. Because low learning rates hinder optimizer convergence, training the optimized CNN model takes longer. Thus, the optimizer maintains a local minimum. When implemented with varied learning rates, the Adam and SGD optimization algorithms produce excellent accuracy, according to empirical study.

The optimized CNN model employs the training data that goes through one cycle during an epoch. I will only use the linked data and the training data respectively only once, per epoch. The number of batches in the database is either 1 or more, where the divided batch size is used in training the model for every epoch. The epoch explains that training instances of each batch are processed one by one. As has been described earlier, the number of data points for each epoch is easily computed by the product of the necessary iterations and the batch size. Alternate epochs can be used for experimentation. The neural network is repeatedly given the same input in this situation. To put it another way, it doesn't fit too well or too terribly. When building a model, we specify the number of epoch parameters before training starts. The most difficult part is deciding how many epochs to use dependent on the batch size. We must determine when the weights converge given the complex neural network's design and the data gathered.

Hyperparameters affect CNN model accuracy developed using the Adam optimizer and different learning rates applied in deep learning. The Adam optimizer is a popular deep learning optimization approach since it controls learning rate for each parameter. The accurateness of the CNN model improved with the Adam optimizer and different learning rates is shown in Table V. We added a learning rate drift to the optimal model after half the epochs. Accuracy data are recorded and the learning rate is adjusted without modifying the kernel. The accuracy variation after five epochs is seen in Table V. The lowered learning rate feature introduced dynamic drift, resulting in accuracy and loss fluctuations at the 5th epoch, as seen in Fig. 4.

TABLE V. OPTIMIZED CNN MODEL PERFORMANCE WITH ADAM OPTIMIZER, LEARNING RATE, AND EPOCHS

No Epochs	Learning rate	Accuracy (%)
1		53.12
2		85.52
3	0.0005	91.50
4		93.81
5		95.59
6		97.92
7		99.07
8	0.0001	99.29
9		99.44
10		99.58

From Tables IV and V, the optimized CNN model's classification accurateness is strongly influenced by the learning rate, so choosing the proper rate is crucial to model performance. Too high or low learning rates reduce optimum CNN model accuracy. The results also revealed that different optimizers might achieve equivalent precision under the correct learning rate and batch size. Thus, the optimizer did not significantly alter model performance. Small batch outcomes are optimal with a lower learning rate.

V. DISCUSSION

The key finding of this study is that a lightweight yet robust CNN architecture, enhanced with dynamic hyperparameter optimization, can achieve state-of-the-art performance in multi-class leaf disease classification. By incorporating three Conv2D blocks and an Inception layer, and systematically tuning learning rates and optimizers, the proposed model attained a classification accuracy of 99.58% on the New Plant Disease Dataset. The results also reveal that a reduced input resolution is sufficient for achieving high accuracy, thereby lowering computational cost without compromising performance. The model demonstrated strong generalization with a precision and recall of 97.53% and an F1-Score of 97.33%, making it effective and scalable for real-world agricultural applications.

In our performance metrics evaluation, the model excels. A record accuracy of 99.58% shows that our model learned from the training data. It had a 97.83% generalization ability with the test data set when evaluated using the feature selection method. The model's accuracy in identifying positive cases and recall relevant instances is 97.53%. Excellent results with a balanced metric coefficient F1 of 97.33%. These metrics together demonstrate our model's efficacy and stability in flagging and categorizing relevant instances for the task.

The robustness evolution of the proposed CNN model, additional experiments were conducted under varying image conditions that mimic real-field environments (shown in Table VI), including changes in lighting, background noise, and image clarity. The test set was augmented with artificially simulated lighting variations—ranging from low exposure (dim) to overexposed (bright) conditions—and with randomized contrast and blur to reflect realistic capture settings in agricultural fields. The model maintained high accuracy under these perturbations, with performance deviations within a 1.2% margin compared to standard testing, demonstrating strong resilience. For example, under low-light augmentation, the model achieved an accuracy of 98.41%, and under high-brightness augmentation, an accuracy of 98.09%, both close to the baseline of 99.02% on the clean dataset.

TABLE VI. ROBUSTNESS EVALUATION OF THE OPTIMIZED CNN MODEL UNDER VARIOUS LIGHTING AND REAL-FIELD CONDITIONS

Condition	Accuracy (%)
Original Test Images	99.02
Low-Light Augmentation	98.41
High-Brightness Augmentation	98.09
Gaussian Blur	97.85
Random Contrast Adjustment	98.27

The dynamic hyperparameter tuning process employed in this study is based on the principles of adaptive learning, where learning parameters are adjusted in response to training dynamics to enhance convergence and model performance. Specifically, the learning rate is scheduled to decay after a predefined number of epochs—most notably after the fifth epoch—enabling coarser updates in the early training phase (promoting exploration) and finer adjustments in later stages (facilitating exploitation). This approach helps avoid premature convergence to suboptimal minima and improves generalization. To further assess the impact of optimization strategies, multiple optimizers—Adam, SGD, and RMSProp—were systematically tested across varying learning rates. The results, as summarized in Tables IV and V, clearly indicate that model performance is highly sensitive to the choice of learning rate and optimizer, with the combination of Adam and a learning rate of 0.0001 yielding the highest accuracy. This empirical strategy for tuning exemplifies a robust, data-driven method that enhances model adaptability and predictive reliability.

VI. CONCLUSION

This paper shows how to create a leaf disease detection model using optimal CNN deep learning with hyperparameters. Thus, the improved CNN model was trained on the New Plant Disease Dataset to obtain 99.58% and 95.59 % at 0.0001 and 0.0005 leaf disease dataset learning rates. Multiple class accuracy averages 99.02% on 38 New Plant Disease Dataset classes. The proposed model outperformed sequential CNN and transfer learning models in accuracy and computation cost. Our model has high generality, training accuracy of 99.58%, test accuracy of 97.83%, precision, recall, and F1-Score of 97.53% and 97.33%, respectively, proving its capability to recognize and categorize. Performance was little affected by the optimizer option. When evaluating CNN model performance with varied learning rate, batch size, and optimizer types, the study found that learning rate had the biggest impact. Empirical investigation shows that SGD learning rate and Adam combination achieve good accuracy. A lower learning rate may improve accuracy, but also restricts batch size, according to the study. The improved model's learning rate drifted after half epochs because variables were adjusted to improve pattern accuracy. The statistics above demonstrate the accuracy difference decreased. The learning rate feature appears after 5 epochs. While the model achieves high accuracy, misclassifications were noted between disease classes with similar visual traits, particularly fungal infections with overlapping symptoms like yellowing or spotting. The use of a reduced input size may contribute to the loss of fine-grained features, affecting class distinction. Additionally, as the dataset consists mainly of clean, centered images, the model may be less effective in real-world settings with complex backgrounds or lighting variations. Despite augmentation efforts, these conditions remain a limitation. Addressing them may require attention mechanisms or testing on real-field datasets.

Future studies could use improved transfer learning and vision transformers to forecast leaf disease. Additionally, future work will include statistical validation through multiple experimental runs such as k-fold cross-validation or repeated hold-out tests. This will allow us to assess the model's robustness and generalization by reporting mean and standard deviation of accuracy, precision, recall, and F1-Score across different data splits. Such statistical rigor would further validate the stability and reliability of the proposed optimized CNN model, especially under varying real-world conditions. The proposed future directions involving transfer learning and vision transformers offer promising avenues for enhancing model performance. Further exploration is encouraged in developing real-time disease detection systems, enabling concurrent identification of multiple infections per leaf, and integrating temporal image data to monitor disease progression over time. These enhancements could significantly improve the practical utility of the model in precision agriculture applications.

CONFLICT OF INTEREST

The authors declare no conflict of interest.

AUTHOR CONTRIBUTIONS

BP: Conceptualization, methodology, software, formal analysis, investigation, programming, writing the original draft, review and editing; AVPK: Investigation, programming, methodology, writing the original draft, editing; and CMR: Formal analysis, investigation, programming, methodology, writing the original draft; all authors had approved the final version.

REFERENCES

[1] C. Dobre, "Upper and lower leaf side detection with machine learning methods," *Sensors*, vol. 22, 26961, 2022.

[2] J. Trivedi, Y. Shamnani, and R. Gajjar, "Plant leaf disease detection using machine learning," *Communications in Computer and Information Science*, pp. 267–276, 2020. doi: 10.1007/978-981-15-7219-7_23

[3] W. Jeon and S. Rhee, "Plant leaf recognition using a convolution neural network," *The International Journal of Fuzzy Logic and Intelligent Systems*, vol. 17, no. 1, pp. 26–34, 2017. doi: 10.5391/ijfis.2017.17.1.26

[4] G. Bhadur and R. Rani, "Agricultural crops disease identification and classification through leaf images using machine learning and deep learning technique: A review," *SSRN Electronic Journal*, 2020.

[5] A. B. Patil, L. Sharma, N. Aochar *et al.*, "A literature review on detection of plant diseases," *European Journal of Molecular & Clinical Medicine*, vol. 7, no. 7, 2020.

[6] G. Sachdeva, P. Singh, and P. Kaur, "Plant leaf disease classification using deep convolutional neural network with Bayesian learning," *Materials Today: Proceedings*, vol. 45, pp. 5584–5590, 2021.

[7] S. M. Hassan, A. K. Maji, M. Jasiński *et al.*, "Identification of plant-leaf diseases using CNN and transfer-learning approach," *Electronics*, vol. 10, no. 12, 1388, 2021.

[8] A. B. Patil and S. D. Khirade, "Plant disease detection using image processing," in *Proc. Int. Conf. Computing Communication Control and Automation (ICCUBEA)*, 2015. doi: 10.1109/iccubea.2015.153

[9] X. E. Pantazi, D. Moshou, and A. A. Tamouridou, "Automated leaf disease detection in different crop species through image features analysis and one class classifiers," *Comput. Electron. Agric.*, vol. 156, pp. 96–104, 2019. doi: 10.1016/j.compag.2018.11.005

[10] J. Liu and X. Wang, "Plant diseases and pests detection based on deep learning: A review," *Plant Methods*, vol. 17, 22, 2021. doi: 10.1186/s13007-021-00722-9

[11] Deepsikha, G. Shrestha, M. Das *et al.*, "Plant disease detection using CNN," in *Proc. ASPCON*, 2020. doi: 10.1109/ASPCON49795.2020.9276722

[12] T. Nguyen and P. D. Hung, "Pest detection on traps using deep convolutional neural networks," in *Proc. ACM Int. Conf.*, 2018. doi: 10.1145/3232651.3232661

[13] X. Guan, "A novel method of plant leaf disease detection based on deep learning and convolutional neural network," in *Proc. ICSP*, 2021. doi: 10.1109/ICSP51882.2021.9408806

[14] A. Dosovitskiy, L. Beyer, A. Kolesnikov *et al.*, "An image is worth 16×16 words: Transformers for image recognition at scale," arXiv preprint, arXiv:2010.11929, 2020.

[15] J.-B. Grill, F. Strub, F. Altché *et al.*, "Bootstrap your own latent: A new approach to self-supervised learning," arXiv preprint, arXiv:2006.07733, 2020.

[16] K. He, H. Fan, Y. Wu *et al.*, "Momentum contrast for unsupervised visual representation learning," arXiv preprint, arXiv:1911.05722, 2019.

[17] R. Sharma, A. Singh, K. Jhanjhi *et al.*, "Plant disease diagnosis and image classification using deep learning," *Computers, Materials & Continua*, vol. 71, no. 2, pp. 2125–2140, 2022.

[18] C. Jackulin and S. Murugavalli, "A comprehensive review on detection of plant disease using machine learning and deep learning approaches," *Measurement: Sensors*, vol. 24, 2022. doi: 10.1016/j.measen.2022.100441

[19] J. Bergstra and Y. Bengio, "Random search for hyper-parameter optimization," *Journal of Machine Learning Research*, vol. 13, no. 10, pp. 281–305, 2012.

[20] D. M. W. Powers, "Evaluation: From precision, recall and F-measure to ROC, informedness, markedness and correlation," *International Journal of Machine Learning Technology*, pp. 37–63, 2011.

[21] P. Sermanet, D. Eigen, X. Zhang *et al.*, "Overfeat: Integrated recognition, localization, and detection using convolutional networks," arXiv preprint, arXiv:1312.6229, 2013.

[22] Vipoooool. New plant disease dataset. [Online]. Available: https://www.kaggle.com/datasets/vipoooool/new-plant-diseases-dataset

[23] A. Krizhevsky, I. Sutskever and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Advances in Neural Information Processing Systems*, pp. 1097–1105, 2012.

[24] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," arXiv preprint, arXiv:1409.1556, 2014.

[25] M. Sandler, A. Howard, M. Zhu *et al.*, "Mobilenetv2: Inverted residuals and linear bottlenecks," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2018, pp. 4510–4520.

[26] C. Szegedy, V. Vanhoucke, S. Ioffe *et al.*, "Rethinking the inception architecture for computer vision," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2016, pp. 2818–2826.

[27] K. He, X. Zhang, S. Ren *et al.*, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2016, pp. 770–778.

[28] M. K. Gurucharan. (Dec 2020). Basic CNN architecture: Explaining 5 layers of convolutional neural network. [Online]. Available: https://www.upgrad.com/blog/basic-cnn-architecture/