IJIE

INTERNATIONAL JOURNAL OF INTEGRATED ENGINEERING
ISSN: 2229-838X e-ISSN: 2600-7916

Vol. 18 No. 4 (2026) 182-198
https://publisher.uthm.edu.my/ojs/index.php/ijie

Early Chilli Disease Detection Based on Leaves Images Using Convolution Neural Network (CNN)

Nurin Jazlina Zulkepple[1], Mazidah Tajjudin[1]*

[1] *Faculty of Electrical Engineering,*
Universiti Teknologi MARA, Shah Alam, 40450, MALAYSIA

*Corresponding Author: mazidah@uitm.edu.my
DOI: https://doi.org/10.30880/ijie.2026.18.04.013

Article Info

Received: 2 October 2025
Accepted: 23 April 2026
Available online: 17 June 2026

Keywords

CNN, chilli leaf, disease detection, Inception-V3, AlexNet

Abstract

Chili is one of the important crops in Malaysia, as it is a main spice used in many local dishes and contributes to the country's economic growth. However, chili plants are prone to diseases, which require early prevention to minimise yield loss. This study presents a CNN-based framework for the early detection of chilli plant diseases based on visible leaf symptoms. Early diagnosis is critical to prevent severe crop loss and reduce the misuse of pesticides. Two deep learning architectures—InceptionV3 and AlexNet—were evaluated using a publicly available Kaggle dataset comprising five categories: healthy, bacterial, fungal, viral, and pest-infected leaves. Data augmentation techniques such as flipping, rotation, and scaling were applied to improve generalization. Both models were trained using transfer learning, with modified final layers to adapt to the specific classification task. InceptionV3 achieved the highest performance with an accuracy of 99.25%, slightly outperforming AlexNet's 98.23%. Evaluation metrics including precision, recall, and F1-score further confirmed InceptionV3's superior capability in extracting relevant features for accurate classification. This system has potential applications in mobile-based diagnosis tools and autonomous field robots. Future work will focus on deploying lightweight models for real-time detection and integrating pesticide recommendation systems for infected plants.

1. Introduction

1.1 Research Background

Chili (*Capsicum spp.*) is one of the most widely cultivated crops in Malaysia due to its high demand in the food industry and economic importance in the agricultural sector. In Malaysia, commercially farmed chili varieties include Red Chili, Bell Pepper, Green Chili, and Kulai Chili. The Malaysian chili market continues to grow, providing a significant source of income for farmers and agro-based industries. According to data provided by Malaysia Ministry of Agriculture and Food Security [1], the production of chilli has increased from 2,8264.4 (mt) in year 2020 to 39,574.4 (mt) in 2023 with production value increasing from RM213,395.91 to RM373,846.21 respectively. However, chili cultivation faces major challenges from plant diseases that affect crop yield and quality. Diseases caused by fungi, bacteria, viruses and pests have adversely affected chili production in almost all parts of the world [2]. The diseases cause symptoms such as leaf yellowing, necrotic spots, and curling, which can be detected through leaf inspection. A recent method of detecting chili disease is done manually by observing visual symptoms that appear on the leaves using naked eyes followed by treatment that involves applying fungicides, pesticides, or biological control agents, but their effectiveness depends on early and accurate diagnosis [3]. In today' s era of Artificial Intelligence (AI) advancement growth, Machine Learning (ML) is commonly used in application of plant disease detection, providing more reliable and automated approaches. Figure 1 illustrates the nested hierarchy of deep learning, where AI encompasses ML, and deep learning (DL) is a subset of ML.

Currently, DL is one of the most used ML-based methods with its characteristic of high levels of abstraction and the ability to automatically learn patterns present in images, becoming today's Fourth Industrial Revolution (IR 4.0) core of technology [5]. According to the research by Gupta et al [6], CNNs can automatically extract features directly from raw image datasets. During training, the network learns important features from batches of images instead of using pre-defined features. Because of this automatic learning ability, CNN has become one of the most accurate models for computer vision tasks, including object detection, classification, and recognition.

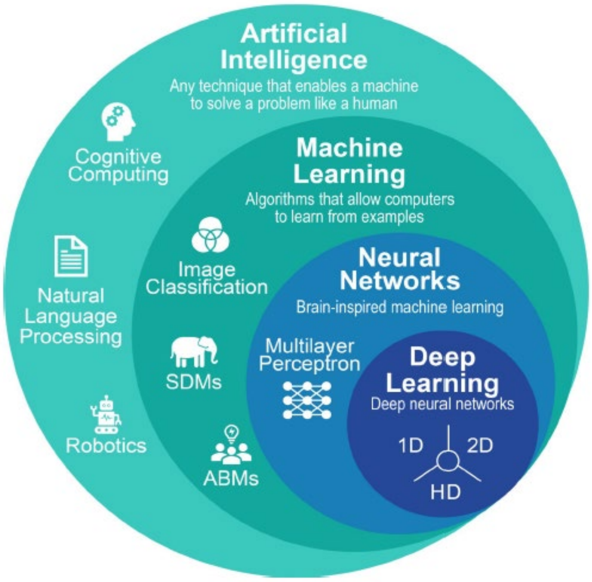

Fig. 1 *Nested hierarchy of deep learning* **[4]**

1.2 Motivation

While chili diseases have long impacted agricultural productivity, effective early detection remains a major challenge due to symptom similarity and reliance on human observation. Motivated by the potential of AI and deep learning in agriculture, this project explores the use of CNNs for accurate image-based classification of chili leaf diseases. The goal is to offer a solution that not only reduces dependency on expert intervention but also supports farmers in making timely decisions. By minimizing crop losses and improving disease management efficiency, this system can help increase overall chili yield and quality, directly contributing to the stability and growth of Malaysia's agro-based economy. Additionally, this project is driven by the opportunity to support smarter, technology-driven agriculture in Malaysia and to gain practical experience in applying deep learning models to real-world problems.

1.3 Problem Statement

Chili cultivation is an essential part of Malaysia's agricultural industry, contributing significantly to the economy. However, chili plants are highly prone to natural diseases that can lead to substantial yield losses if not detected early. Different diseases exhibit distinct symptoms on the leaves, and in some cases, multiple diseases may infect a plant simultaneously. Farmers must accurately identify the type of disease to apply the correct treatment, whether through fertilizers, pesticides, or plant removal. However, manual disease identification is challenging due to symptom similarities, environmental conditions, and human limitations.

A recent method of detecting chili disease is done manually by observing visual symptoms that appear on the leaves using naked eyes followed by treatment that involves applying fungicides, pesticides, or biological control agents, but their effectiveness depends on early and accurate diagnosis. The similarity in morphological features among different plant diseases further complicates manual identification, leading to misdiagnosis and ineffective treatment [3]. With growing agricultural demands and the need for sustainable and precise farming practices, traditional approaches to plant disease detection are no longer sufficient.

Int. Journal of Integrated Engineering Vol. 18 No. 4 (2026) p. 182-198

1.4 Objectives

This study is conducted with the following objectives to address the early detection of chili plant diseases through image analysis and to evaluate the effectiveness of different convolutional neural network architectures:

i. To develop a CNN model for early chili disease detection based on visual symptoms seen on leaves.
ii. To compare the overall performance of InceptionV3 and AlexNet architectures in recognizing chilli leaves symptom image recognition.

1.5 Significance of Study

The early detection of chili plant diseases plays an important role in keeping crops in a healthy state, reducing yield losses, and improving overall agricultural productivity. By leveraging deep learning techniques for automated disease detection, this project offers significant advancement over recent manual methods. This method enables farmers to take immediate corrective actions, preventing the spread of infections and minimizing economic losses. This contributes to more sustainable farming practices, reducing excessive usage of pesticides and fertilizers, which can have negative environmental impacts.

In addition to benefiting farmers, this project has broader implications for the agriculture industry and food security. Chili is a high-demand crop in Malaysia, and disease outbreaks can lead to supply shortages and increased market prices. By integrating AI-based solutions into agriculture, this research supports the modernization of farming techniques, helping farmers transition toward precision agriculture. The adoption of automated disease detection systems can enhance decision-making, optimize resource usage, and improve overall crop management efficiency.

From a technological perspective, this project contributes to the growing field of deep learning applications in agriculture. CNN implementation for image classification shows the potential of AI capability to solve real-world agricultural challenges. Additionally, the development of an automated disease detection model could be integrated into mobile applications or IoT-based agricultural systems, making it more accessible to farmers regardless of their technical expertise.

1.6 Project Scope

The scope of this project includes the development of an early detection system of chili plant diseases through image identification from the leaf symptoms. A crucial part of this project involves data collection and database development, where suitable and enough data in a dataset of chili leaf images will be compiled. These images will consist of both healthy and infected leaves, sourced from publicly available datasets from Kaggle database and supplemented with field-collected samples. This variation of dataset is essential to ensure the model's robustness in recognizing different disease symptoms under different environmental and surroundings conditions.

For model development, this project implements and compares two CNN architectures: InceptionV3 and AlexNet. InceptionV3, a deeper and more advanced model, is known for its ability to extract multi-scale features using inception modules, making it highly effective for complex image classification tasks. On the other hand, AlexNet is a simpler, earlier CNN architecture that laid the foundation for modern deep learning in vision tasks. By comparing these two models, this project aims to evaluate the trade-offs between computational efficiency and classification performance. Key performance metrics such as accuracy, precision, recall, and F1-score will be analyzed to determine which model is better suited for early chili disease detection. To assess generalization, both models will be trained and tested using a structured chili leaf image dataset. The comparison will help identify the most reliable and practical model for future deployment in mobile applications or automated agricultural systems.

2. Literature Review

2.1 Deep Learning

According to research by Mukhamediev et.al [7], the field of artificial intelligence (AI) has been extensively developed and actively discussed within the scientific community. These studies explore both the theoretical foundations of AI technologies and their practical applications across various sectors of society. Within AI, machine learning (ML) represents a set of techniques that enable systems to predict new data properties based on patterns learned from training data. A more specialized area within ML is deep learning (DL), which has gained increasing research attention in recent years. Kumari et.al [8] in their study also highlighted that Emerging AI, robotics, the Internet of Things (IoT), big data analytics, and ML have advanced to the point where machines can perform tasks in ways similar to humans. In particular, the concepts of AI, ML, and DL are becoming more prominent in the financial sector and are attracting growing interest from the research community.

Mohd Noor and Ige [9] in their research stated that Deep Learning, a subset of artificial intelligence, is a data-driven approach that leverages multiple layers of interconnected neurons to automatically learn complex patterns and representations from raw input data. This powerful learning capability has made deep learning a core technology behind many groundbreaking innovations and solutions. It continues to revolutionize how businesses operate and how complex problems are approached and solved. Convolutional Neural Network (CNN) is one of three key types of deep supervised learning models that are trained using labeled datasets. The learning process involves computing prediction errors through a loss function and iteratively adjusting model weights to minimize those errors.

Yan et al. [10] in their study state that CNNs are a type of feedforward neural network that integrate convolutional operations within a deep architecture, drawing inspiration from the biological receptive field mechanism. A typical CNN architecture comprises multiple hidden layers, including convolutional layers, pooling layers, fully connected layers, and activation functions. CNNs are primarily designed for processing two-dimensional data, making them particularly effective for image recognition, classification, and segmentation tasks.

2.2 Application of Deep Learning in Plant Disease Detection

A study by Dalal & Mittal [11] stated that Deep Learning techniques have significantly transformed the agricultural sector by offering accurate and efficient solutions for tasks such as disease detection, crop monitoring, and yield prediction. Models like CNNs, Recurrent Neural Networks (RNNs), Generative Adversarial Networks (GANs) and transfer learning are widely applied in agricultural applications, including plant disease detection, image classification, crop surveillance, and weed identification using images or videos.

AlexNet continues to serve as a reliable baseline in recent plant disease detection research due to its simple structure and relatively fast training time. A study [12] proposed HOG and LBP weighted fusion to extract image features, a tomato leaf disease recognition model based on the AlexNet model using 8 types of tomato leaf diseases and healthy leaves with a total of 16563 images. The improvised AlexNet achieved accuracy of 98.72 % and a balanced F-score of 99.4 %. Transfer learning techniques are used pretrained on PlantVillage dataset, and the training took 200 epoches. The limitation of the study is that it was conducted in an experimental environment where the background was removed, which may affect the results in real-life settings

Likewise, Y. Jawad et.al [13] used five DL models including AlexNet, VGG-16 and ResNet-50 for classifying five different types of dates fruit and its maturity stage. AlexNet achieved an accuracy of 97.33% for fruit types and 98.17% for the maturity stage. Autonomous robotic vehicle front and top cameras are used to localize the quick response (QR)-labeled palm trees using canny edge detection and date bunch detection and capturing. The captured images are transferred to User Interface (UI) using Firebase for farmers monitoring purposes. The developed UI is practical and helps farmers easily classify images and make harvesting decisions. This improves usability and supports decision-making in the field. The smart harvesting robot works in natural environments, which is useful for real-life farming. It can help improve fruit yield and support the global supply chain. However, the system only focuses on one fruit and may not work well with other crops. The trained model was also tested using images containing more than one date bunch to evaluate its detection performance model successfully detected two date bunches with a confidence of 91% and the third with a confidence of 89%. The proposed autonomous system was tested in real field conditions, including date bunches covered with protective bags.The robotic vehicle was able to successfully detect the covered date bunches from the images captured by the rover. There are no mention of how well the UI works for farmers with low digital skills or different languages. These factors may limit the system's use in wider farming communities.

InceptionV3 has been widely adopted in recent agricultural studies due to its advanced multi-path architecture that captures both fine and coarse image features. According to Hukkery et.al [14], InceptionV3 achieved an accuracy of 93.8% accuracy when used for diagnosing disease-infected leaves using 38 different classes and 14 different plant species in total. The total dataset used is 24,305 snapshots that are obtained from Kaggle, and the models were trained for 30 epochs using transfer learning. Even though the total number of data is large, dividing it into classes results in a small number of samples per class. It is recommended to work with larger and more diverse leaf datasets using recent algorithms, as this can help improve prediction accuracy, especially in challenging environments.

A study conducted by Alam et al. [15] on developing state-of-art deep learning models for detecting and localizing ripe tomatoes in a greenhouse using five deep learning models including DenseNet121 and InceptionV3. Total data used in the study is 895 images where this base data is split into 80% for training and 20% for testing while for validation process, the data used is randomly picked 100 augmented images from base data. The InceptionV3-based model has slightly lower performance compared to other models, but still achieves high precision and recall values, especially for "Unripe" tomatoes, with 98% recall score. However, the dataset size of 895 images is relatively small, which may affect the model's ability to generalize in real-world scenarios. Using only 100 augmented images from the same dataset for validation could introduce bias and does not truly reflect model performance on unseen data. Testing was limited to greenhouse conditions, so its effectiveness in uncontrolled environments remains unproven.

Soujanya and Jabez [16] in their study proposed an improved AlexNet to classify plant disease at an early stage using PlantVillage dataset. The proposed method adds to the conventional AlexNet the deconvolution layer

Int. Journal of Integrated Engineering Vol. 18 No. 4 (2026) p. 182-198

to decrease the percentage of parameters and the cost of computation, classifying images with a fully connected layer. This study utilizes the whole dataset and is splitted into 80% for training and 20% for testing. Result shows the improvised AlexNet obtains 96.55% score overall accuracy on augmented validation dataset. This research shows that AlexNet still has potential in complex applications when improvements are made, even though it is one of the earliest and most basic CNN architectures. These studies focus on categorizing symptoms into specific classes. However, in this paper, the research will focus on the main categories: healthy, bacterial, fungal, viral, and pest. All symptoms, such as upward curling, bacterial spots, and whitefly infestations, are treated as subcategories under these main classes.

2.3 Chili Disease Detection Using Deep Learning

Rozlan and Hanafi [17] conducted a study on efficacy of VGG16, InceptionV3 and EfficientNetB0 architectures in classifying 3 chilli plant disease; upward curling, mosaic, and bacterial spot using 3,000 images collected from field environments in Selangor. The input image for InceptionV3 model is set as 299 X 299 X 3. The experiment was conducted two times using original images and augmented images including shear, zoom and horizontal flip. Result shows that InceptionV3 achieved highest accuracy on augmented images with a score of 98.83%. This study uses a good approach by collecting data in an uncontrolled environment with various lighting conditions and real-life views of chilli leaves, which helps improve the model's robustness for real-world applications. However, only three disease classes were selected in this research, despite the existence of many other chilli plant diseases that could have been included.

Adnan et al. [18] conducted research aims to classify chilli diseases such as yellowing, fungal infection, and whitefly infestation using an ensemble technique approach, utilizing pretrained CNN models including MobileNetV2, ResNet152, EfficientNetB7, EfficientNetV2B0, InceptionV3, ResNet50, DenseNet121, and DenseNet201. The dataset consists of 800 images sourced from multiple websites, varying in form, size, and colour. The models were customized by unfreezing the final 10 layers and replacing them with dense layers and custom pooling layers. Results show that InceptionV3 demonstrated strong performance across all evaluation metrics, particularly in accuracy, achieving a score of 95.68% using the RMSProp optimizer, outperforming both ResNet152 and ResNet50. One major limitation of the study is the small dataset size, which may significantly impact the generalizability of the results. Limited number of images may reduce models' ability to perform accurately on unseen, real-world samples. This raises concerns about the model's robustness when applied to more diverse or complex data sets in practical agricultural settings.

In addition to architectures like InceptionV3 and EfficientNet, AlexNet has also been applied specifically to chilli plant disease detection as in research conducted by Naik et.al [19]. Chili leaves that have been infected by five different diseases; up curl, down curl, Cercospora leaf spot, Geminivirus, and yellow leaf were plucked from a field located in Andhra Pradesh, India. The leaves were then set in a blue background, and the images were captured using Sony 7.1MP digital camera. Data augmentation techniques used are position augmentation including scaling, cropping, flipping, padding, rotation, translation, and affine transformation, and colour augmentation including brightness, contrast, saturation, and hue. Results for AlexNet model achieved 97.65% accuracy when pretrained using 0.0001 learning rate and SGDM optimizer. This research presents a well-structured setup for comparing 12 CNN models in classifying chilli plant diseases, as the researchers built their own dataset by collecting real-life images, equally dividing them into five classes, and applying a variety of data augmentation techniques.

3. Research Methodology

3.1 Introduction

Figure 2 shows the project workflow, which begins with collecting the dataset in image form, followed by preprocessing the dataset, training and evaluating model performance using the training and evaluation dataset, testing performance using the test dataset, comparing the performance of two model architectures using evaluation metrics, and selecting the best-performing model for further implementation.

The experiments in this study were carried out on a desktop machine running Windows 11, equipped with an Intel Core i5-10400 processor, 8 GB of RAM, and an NVIDIA GPU. All model development and training were conducted using TensorFlow 2.13, a leading deep learning framework widely used for large-scale machine learning applications. TensorFlow is an open-source deep learning framework developed by the Google Brain team and released in 2015. It provides a comprehensive ecosystem for building and deploying machine learning models across a range of platforms, including desktops, mobile devices, and cloud-based systems. TensorFlow supports both low-level operations for fine-grained model control and high-level APIs for ease of use, making it suitable for both research and production environments. Its computation is based on data flow graphs, which allow complex operations to be represented as a series of nodes and edges, facilitating efficient execution on CPUs, GPUs, or TPUs.

The implementation was done in Python within a Jupyter Notebook environment managed via Anaconda Navigator, utilizing TensorFlow's integrated Keras API for building, image preprocessing, training, evaluating and testing the neural network models. Keras is a high-level neural network API that runs on top of deep learning frameworks such as TensorFlow. It was initially developed as an independent project by François Chollet and later integrated into TensorFlow as its official high-level API. Keras simplifies the process of building and training deep learning models by providing user-friendly interfaces and modular components such as layers, optimizers, and loss functions. Its intuitive design and minimal code requirements make it especially suitable for beginners and rapid prototyping, while still offering advanced functionalities for experienced developers through TensorFlow's backend.

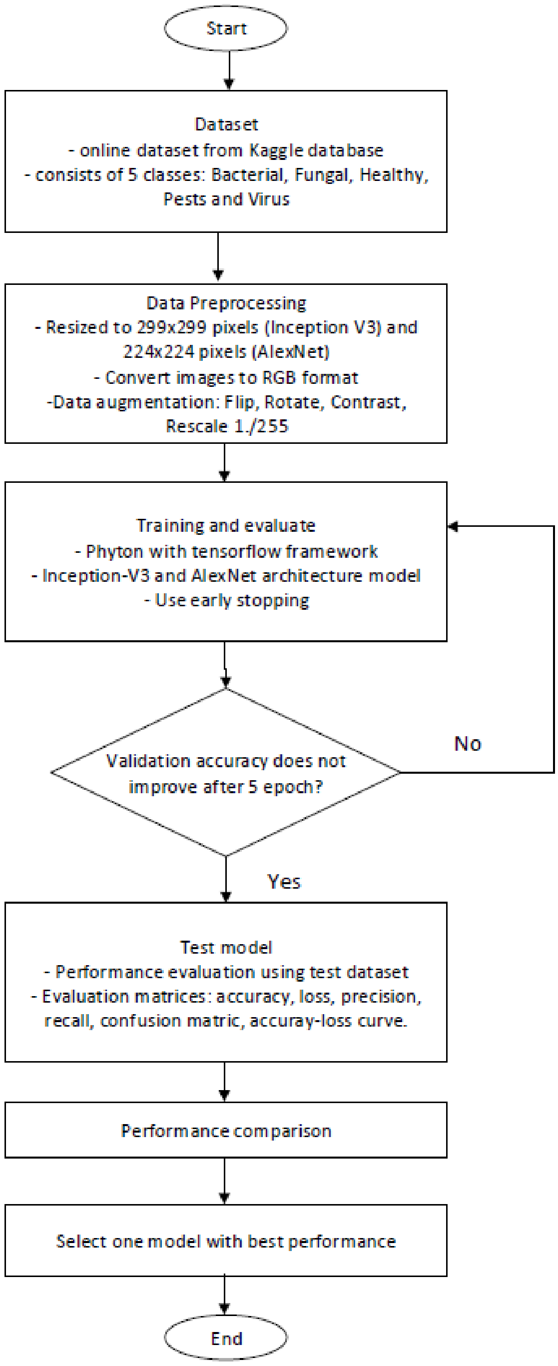

Fig. 2 *Project flowchart*

3.2 Data Collection and Preparation

The dataset used in this study is a combination of two datasets that are available on Kaggle online database namely Chili-Dataset [20] and chilli_leaf_dataset [21]. The data was downloaded into a local folder and combined according to their classes: Bacterial, Fungal, Healthy, Pests, and Virus. Each class was then split into three different sets: train, validate, and test. Initially, the dataset was provided in two separate folders: train and test. The images from the train folder were further divided into 80% for training and 20% for validation. Meanwhile, the test folder was kept as an independent dataset and used only for the final testing process. In total, there are 4,899 images for training, 1,222 images for validation, and 1,470 images for testing.

Figure 3 displays three random samples from each of the five main classes—healthy, bacterial, fungal, viral, and pest—selected from the training dataset. These images serve as the primary input for training the classification model. To enhance model performance and improve generalization, the training images will undergo various augmentation techniques such as rotation, flipping, and contrast adjustment. This helps increase the diversity of the training data and reduce the risk of overfitting. Figure 4 presents three random samples from each of the five classes drawn from the validation dataset. These images are not seen by the model during training and are used to evaluate and fine-tune the model's performance, helping to ensure it generalizes well to new data. Figure 5 shows three random samples from each of the five main classes taken from the test dataset. These images are used exclusively to assess the final model's accuracy and generalization capability on completely unseen data, serving as the ultimate performance evaluation.

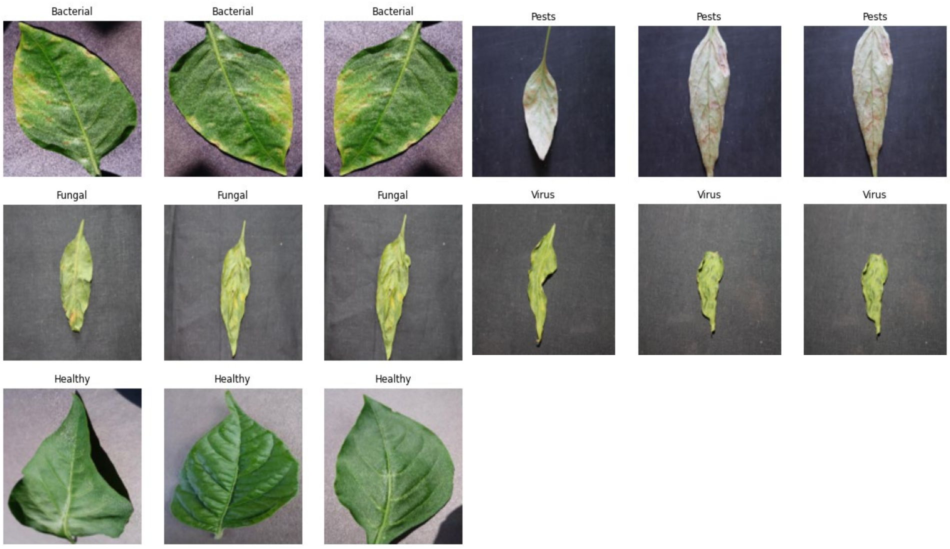

Fig. 3 *Samples of train dataset from 5 classes*

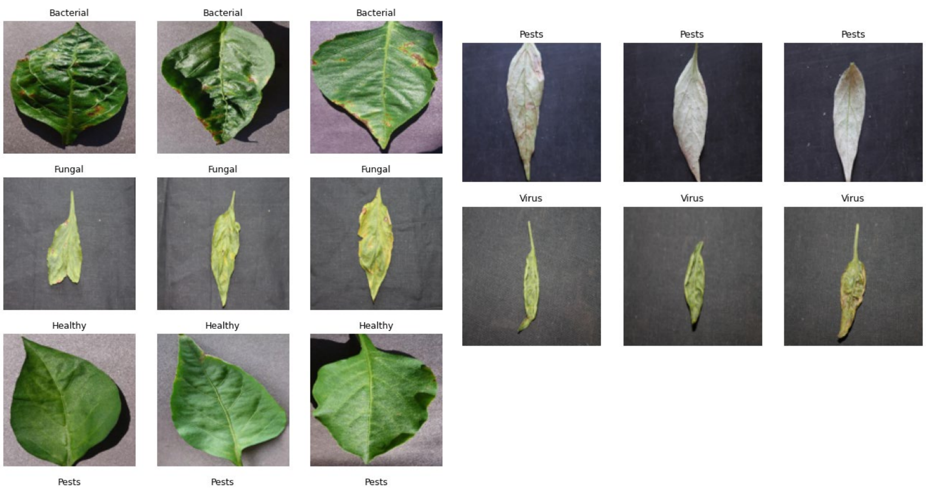

Fig. 4 *Samples of train dataset from 5 classes*

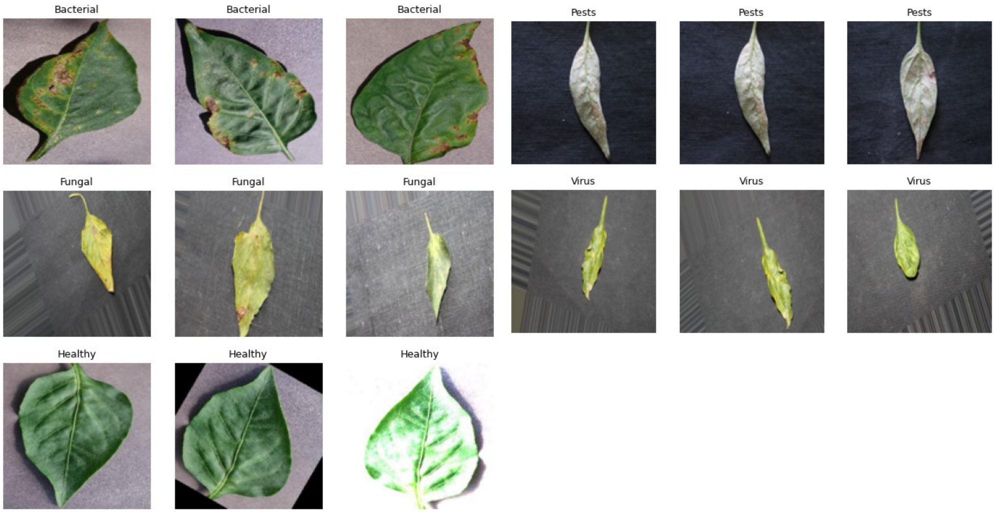

Fig. 5 *Samples of test dataset from 5 classes*

3.2.1 Data Pre-Processing

Inception V3 is trained using transfer learning because it was already trained on a big dataset called ImageNet using high-resolution images 299×299. Its deep and complex structure makes it good for fine-tuning when we want to use it on a new dataset. Some studies, like in medical image classification, show that fine-tuning Inception V3 gives better results than training it from the beginning [22]. On the other hand, AlexNet is an older model with a simpler structure of five convolutional layers and three fully connected layers, is trained from scratch due to its simpler architecture and it uses smaller image sizes which is 224×224. Figure 6 shows sample images after augmentation are applied on train data.

The data augmentation in the code is used to increase the variety of training images without collecting new data. This helps improve the model's performance and reduce overfitting. The process is done using three techniques: random horizontal flip, random rotation, and random contrast adjustment. The *RandomFlip*('horizontal') layer flips the image from left to right randomly. This helps the model learn that the object is the same even if it appears flipped. The *RandomRotation*(0.2) layer rotates the image by up to ±20% of 360 degrees (about ±72 degrees), which helps the model recognize objects even if they are slightly rotated. The *RandomContrast*(factor=0.2) layer changes the contrast of the image randomly, which makes the model more robust to changes in lighting.

One image is taken from the training dataset, and the augmentation function is applied to it nine times. Each time, the image may look slightly different because the changes are random. These nine versions are displayed in a 3×3 grid. This shows how a single image can be changed in many ways, which helps the model train better on different-looking images. This process effectively expanded the dataset, enhancing the model's ability to adapt to variations in features and improving its overall robustness [23].

Fig. 6 *Sample of augmented images*

3.3 Model Selection and Training

In this study, InceptionV3 and AlexNet have been selected as the Convolutional Neural Network (CNN) architectures to classify chili plant diseases.Figure 7 visualises the architecture of Inception V3 model meanwhile Figure 8 shows the architecture of AlexNet model.

Inception V3 architecture consists of a deep CNN model developed by Google as an enhancement to earlier Inception networks or known as GoogleNet. It consists of 42 layers integrated by: Convolution Layers, Avg Pool, MaxPool, Concat, DropOut, Fully Connected and Softmax [24]. The architecture is known for its inception modules, which allow the network to capture features at multiple scales within the same layer. Inception V3 has been widely used in large-scale image classification benchmarks such as ImageNet and provides a strong balance between depth, performance, and computational efficiency.

On the other hand, AlexNet, introduced by Alex Krizhevsky in 2012 [26], is a deep learning model based on the convolutional neural network (CNN) architecture as illustrated in Figure 8, AlexNet comprises a total of eight layers and can classify images across a wide range of categories with high accuracy. The architecture includes five convolutional layers followed by three fully connected layers, forming the core structure of the CNN model. These convolutional layers are responsible for automatically extracting important features from input images, such as edges, textures, and shapes, while the fully connected layers perform the final classification based on the learned features. Additionally, AlexNet uses ReLU activation functions to introduce non-linearity, max-pooling layers to reduce spatial dimensions, and dropout to minimize overfitting during training. The model gained significant attention after winning the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) in 2012, demonstrating the potential of deep CNNs in computer vision tasks.

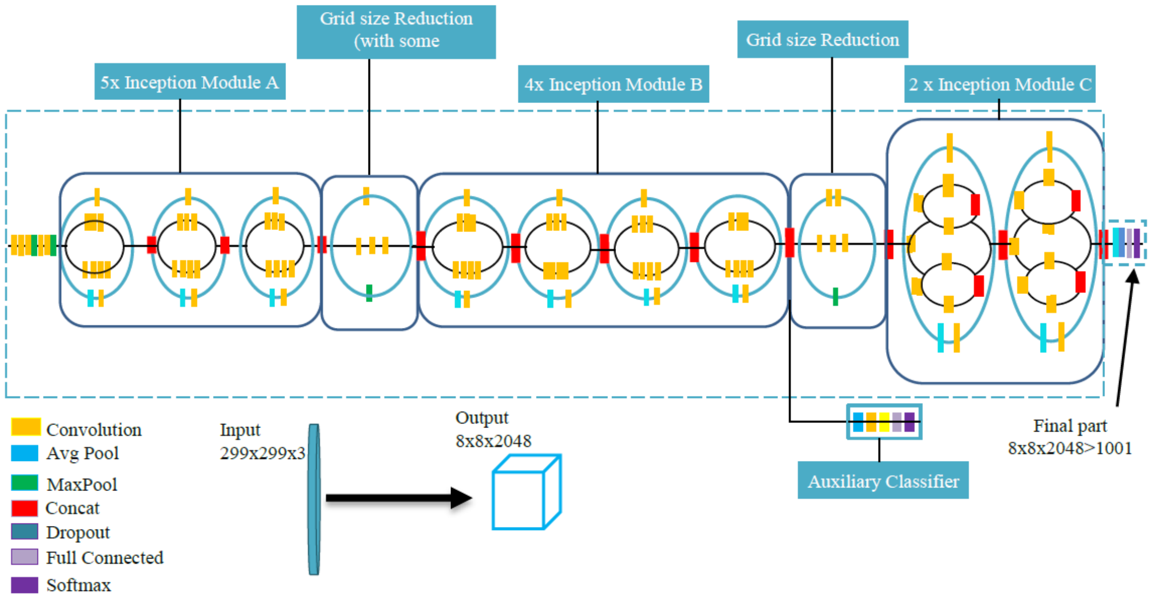

Fig. 7 *Illustration of InceptionV3 model [24]*

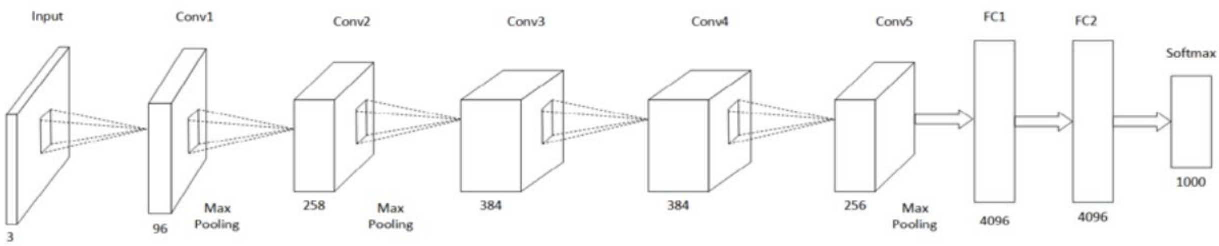

Fig. 8 *Illustration of AlexNet model [25]*

3.3.1 Model Evaluation

The models are evaluated on the test set to evaluate their classification accuracy, precision, recall, and F1-score. Confusion matrices are generated to analyses the classification performance for each disease category. The evaluation results compare the accuracy of image classification between AlexNet and InceptionV3 where only one model with higher performance is selected in this study. For evaluation, sklearn.metrics' library will be used to calculate models' accuracy, precision, recall, and F1-score meanwhile confusion matrix can be visualized in Phyton using 'matplotlib' library for graphical representation. Specification of the above-mentioned metrics are:

- True positive (TP): Leaves which have disease and correctly predicted as diseased.
- True negative (TN): Leaves which do not have disease and correctly predicted as not diseased.
- False positive (FP): Leaves which do not have disease but falsely predicted as diseased.
- False negative (FN): Leaves which have disease but falsely predicted as not diseased.
- Precision: The proportion of true positives to those that the model predicted as positive.

$$Precision = \frac{TP}{TP + FP} \tag{1}$$

- Recall: To measure the model's ability to locate all the positives.

$$Recall = \frac{TP}{TP + FN} \tag{2}$$

- F1-score: The F1-score, which ranges from 0 to 1, can be used to calculate testing accuracy.

$$F1 = 2 \times \frac{Precision \times Recall}{Precision + Error} \tag{3}$$

- Confusion Matrix: Table 1 shows the structure of the confusion matrix, which is used to evaluate the overall performance of the model.

Table 1 *Confusion matrix structure*

	Predicted Positive	Predicted Negative
Actual Positive	True Positive (TP)	False Negative (FN)
Actual Negative	False Positive (FP)	True Negative (TN)

4. Results and Discussion

4.1 Introduction

This chapter presents the results obtained from the implementation and evaluation of the two CNN models—InceptionV3 and AlexNet—used for early chili leaf disease detection. The performance of both models is assessed based on key evaluation metrics, including accuracy, precision, recall, F1-score, and loss. In addition, training and validation performance over epochs are analysed using accuracy-loss curves, and confusion matrices are used to examine class-wise predictions. The results are then discussed in relation to the research objectives, highlighting the strengths, limitations, and comparative performance of each model. This chapter aims to determine the most suitable model for accurate and efficient chili disease classification and its potential for real-world agricultural deployment.

4.2 Model Performance and Evaluation

During the initial model development of InceptionV3, dropout values of 0.5 and 0.3 were applied in the InceptionV3 architecture to prevent overfitting by randomly deactivating 50% and 30% of neurons during training. However, this resulted in significantly longer training times, with the model taking approximately 7 hours to complete training. The high dropout rate slowed down the learning process as too many neurons were being ignored during each training iteration, limiting the model's capacity to effectively learn patterns in the data. After adjusting the dropout rate to 0.2, the training time was drastically reduced to just 1.5 hours, and the model achieved better classification performance. This improvement suggests that a lower dropout rate maintained a better balance between regularization and model capacity, allowing the network to converge more efficiently while still preventing overfitting. This finding is consistent with the results reported by Harris & Andrews [27] that model performance decreases when the dropout rate is set too high, particularly above 0.75. Excessive dropout can lead to underfitting because the network becomes too restricted to learn meaningful patterns from the data, resulting in unstable training and reduced accuracy.

In the initial training phase of the AlexNet model, a batch size of 64 was used along with a fixed number of epochs to reduce training time. However, the results were unsatisfactory, as reflected in the confusion matrix in Figure 9 which showed a significant number of misclassifications across all classes. The larger batch size may have

Int. Journal of Integrated Engineering Vol. 18 No. 4 (2026) p. 182-198

led to less stable gradient updates, limiting the model's ability to learn subtle class differences effectively. In the subsequent trial, the batch size was reduced to 32, and early stopping with a patience value of 7 was introduced to prevent overfitting and allow the model to stop training once performance plateaued. This configuration resulted in a much more stable and accurate system. Despite the training taking approximately 45 hours, the outcome demonstrated significantly better model generalization and reliability.

The average training time for AlexNet model was approximately 45 minutes, meanwhile InceptionV3 was approximately 1.5 hours. From Figure 10, the training accuracy of Inception-V3 steadily increases over the epochs after fine-tune. The validation accuracy experiences a slight drop around epoch 4 but improves again by epoch 6. Meanwhile, the training loss gradually decreases, and the validation loss follows a similar trend, with a slight increase near Epoch 4 before decreasing again. After that, the model is fine-tuned to continue training. The base of the pretrained InceptionV3 model is unfrozen, allowing its layers to be updated during training. InceptionV3 contains 311 layers in its base model, but fine-tuning is applied starting from layer 250 onwards, meaning the earlier layers remain frozen. After fine-tuning, the accuracy increases with a steeper gradient before gradually plateauing. The same trend is observed for both training and validation loss, which decreases more rapidly after fine-tuning and then levels off. This indicates that the model is performing well and learning effectively from the fine-tuning process.

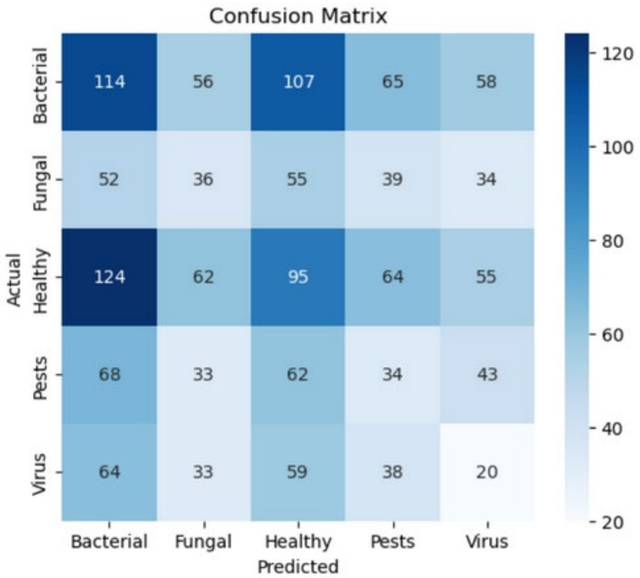

Fig. 9 *Confusion matrix of AlexNet initial training*

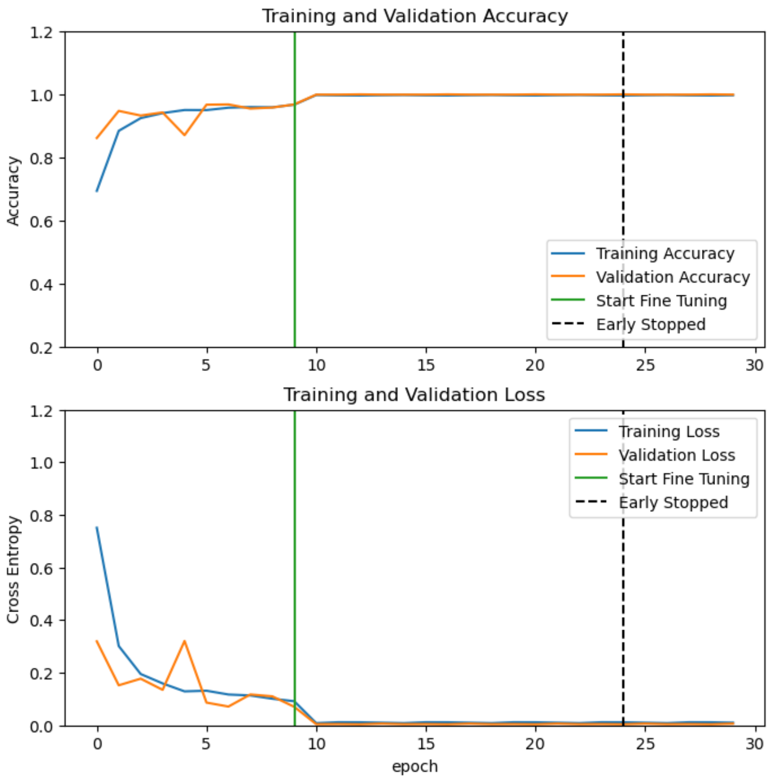

Fig. 10 *Accuracy-loss for training and validation over epoch for inception-V3 after fine-tune*

From the accuracy-loss graph for AlexNet model in Figure 11, training accuracy increases quickly and reaches nearly 100% by around epoch 10. Validation accuracy starts lower but rises steeply, stabilizing around 95–98% after epoch 5. There are slight fluctuations in validation accuracy, but no major drops. The gap between training and validation accuracy is small, which indicates that the model performs well on unseen data. There are slight fluctuations in validation accuracy, but no major drops. Meanwhile both training and validation loss decreased sharply in the early epoch. The training loss continues to drop and remains low, while the validation loss fluctuates slightly but stays relatively stable after epoch 5. These patterns confirm that the model is learning effectively without signs of overfitting or underfitting.

Table 2 shows the accuracy and loss scores of InceptionV3 and AlexNet across training, validation, and test datasets. InceptionV3 achieved consistent performance, with training and validation accuracy both at 96.73%, and a high-test accuracy of 99.25%. It also maintained low loss values across all datasets, indicating good generalization. AlexNet achieved a higher training accuracy of 99.63% and slightly better validation accuracy of 97.79%, with a test accuracy of 94.71%. While its test loss, 0.1500, was higher than that of InceptionV3 0.0651, the overall accuracy remained high across all phases. This suggests that AlexNet also generalized well, though its performance on unseen test data was slightly lower than InceptionV3.

A detailed comparison of the five deep learning models, based on overall accuracy, precision, recall, and F1-score, is presented in Table 3. Inception-V3 consistently outperforms AlexNet across all measures. It recorded the highest accuracy of 0.99251, along with precision, recall, and F1-score values all at 0.99251. On the other hand, AlexNet achieved slightly lower values, with accuracy at 0.9471, precision at 0.9772, recall at 0.9762, and F1-score also at 0.9762.

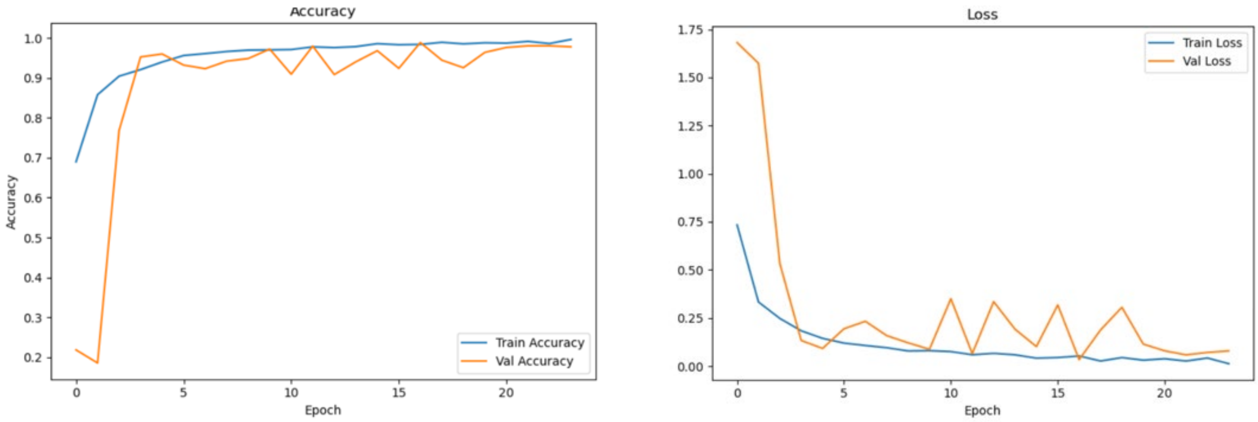

Fig. 11 *Accuracy-loss for training and validation over epoch for AlexNet*

Table 2 *Accuracy and loss score*

Model		Train	Validate	Test
Inception V3	Accuracy	0.9673	0.9673	0.9925
	Loss	0.0914	0.0704	0.0651
AlexNet	Accuracy	0.9963	0.9779	0.9471
	Loss	0.0126	0.0790	0.1500

Table 3 *Accuracy, precision, recall and F1-score for inception-V3*

Matrices	Train	Validate
Accuracy	0.99251	0.9471
Precision	0.99251	0.9772
Recall	0.99251	0.9762
F1-Score	0.99251	0.9762

Figure 12 shows the confusion matrix of the Inception-V3 model, demonstrating its high classification performance across all five classes. The model correctly classified almost all instances, with only a single misclassification observed. Specifically, out of 400 Bacterial images, 389 were correctly predicted, while 11 were misclassified as Healthy. The remaining classes, Fungal, Healthy, Pests, and Virus—were predicted with 100% accuracy, with all respective images falling exactly on the diagonal of the confusion matrix. This indicates that the Inception-V3 model is highly effective in distinguishing between the different disease types and healthy conditions in chilli leaf images.

Figure 13 presents the confusion matrix for the AlexNet model, which, although still performing well, showed slightly lower classification accuracy compared to Inception-V3. The Bacterial class had the highest number of misclassifications, with 28 out of 400 images predicted as Healthy. The Fungal class saw 4 images misclassified as Viral. Additionally, 3 images from the Healthy class were mistakenly classified as Bacterial. The Pest and Virus classes, however, were predicted with perfect accuracy, indicating that AlexNet can effectively identify these two classes. The presence of more off-diagonal values suggests that AlexNet is less accurate in distinguishing between certain disease types than Inception-V3.

The wrong prediction may be caused by the similarity of the photo shape between the Healthy and Bacterial classes, as shown in Figure 14. Since both classes share similar leaf shapes, the model may struggle to distinguish between them. In some cases, bacterial infections at an early stage may not produce obvious symptoms, causing the leaf to appear visually similar to a healthy leaf. As a result, the deep learning model may incorrectly classify the images due to the overlapping visual characteristics between these two classes. Similar challenges have been reported in plant disease classification studies [17] where the number of true positive predictions was lower compared to other diseases because their symptoms were visually similar to upward leaf curl disease.

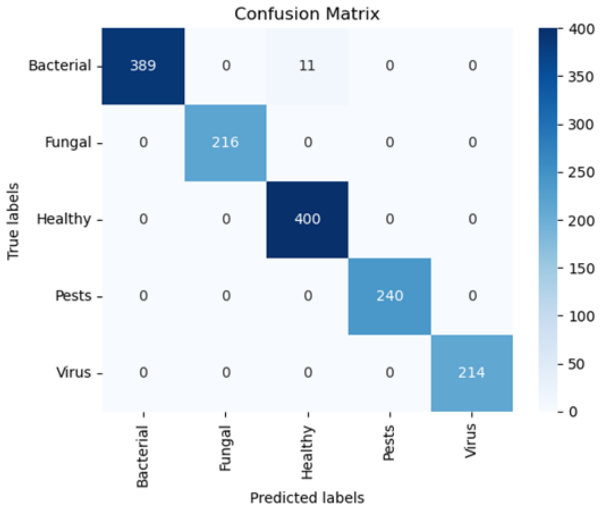

Fig. 12 *Confusion matrix of inception-V3 model*

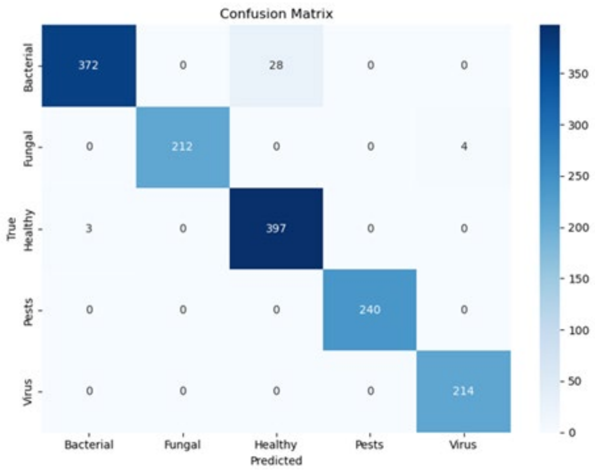

Fig. 13 *Confusion matrix of AlexNet model*

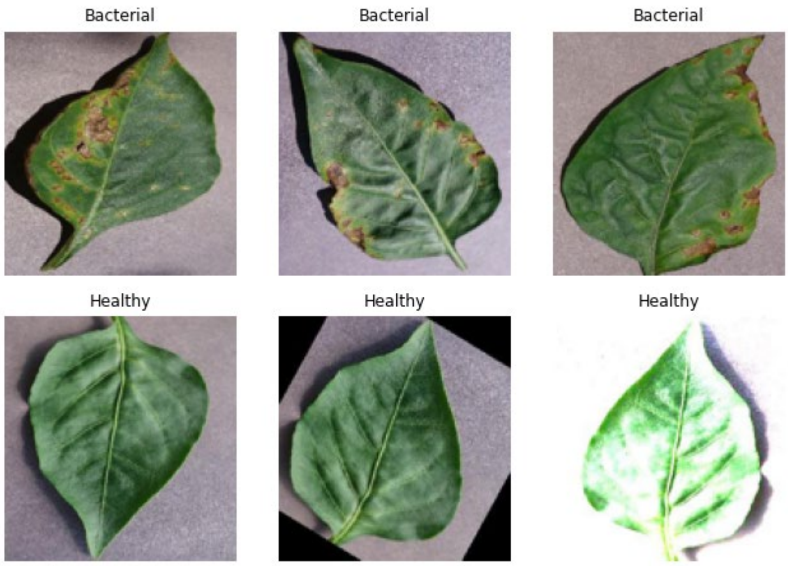

Fig. 14 *Sample of healthy and bacterial data images*

4.3 Comparison and Discussion

This section compares the performance of InceptionV3 and AlexNet in detecting chili leaf diseases based on image classification. Overall, both models demonstrated strong performance; however, InceptionV3 consistently outperformed AlexNet in terms of test accuracy, loss, and evaluation metrics such as precision, recall, and F1-score. InceptionV3 achieved a test accuracy of 99.25% with a lower loss value, indicating better generalization to unseen data. This can be attributed to its deeper architecture and the use of inception modules, which allow the model to extract features at multiple scales. In contrast, AlexNet, despite achieving high training and validation accuracy, showed slightly lower test accuracy at 94.71%, suggesting it is slightly less effective at capturing complex or subtle disease features. Additionally, the confusion matrix showed that both models occasionally misclassified between classes with similar visual symptoms, particularly between the 'Healthy' and 'Bacterial' classes. Although the difference is narrow, AlexNet made more incorrect predictions compared to InceptionV3, indicating that InceptionV3 was slightly more effective at distinguishing between visually similar disease categories. This finding supports previous studies which reported that InceptionV3 outperforms other deep learning models such as VGG-16 and ResNet50 in image classification tasks [28], [29]. Based on the overall results and its superior generalization capability, InceptionV3 is selected as the preferred model for the chili early disease detection task, offering greater reliability and suitability for practical deployment in agricultural applications.

5. Conclusion and Recommendation

5.1 Conclusion

This study aimed to develop a CNN-based model for early detection of chili plant diseases using leaf images and to compare the performance of two deep learning architectures, AlexNet and InceptionV3. The experimental results demonstrated that both models could classify chilli leaf diseases; however, InceptionV3 consistently achieved better performance, with higher accuracy, precision, recall, and F1-score compared to AlexNet. The confusion matrix analysis further confirmed that InceptionV3 reduced misclassification between visually similar classes such as Healthy and Bacterial leaves. These findings indicate that deeper architectures with advanced feature extraction capabilities are more effective for plant disease image classification tasks. The proposed framework contributes to the development of automated plant disease detection systems, which can assist farmers in identifying diseases at an early stage and taking timely preventive actions to reduce crop losses. Such systems also have potential applications in mobile-based diagnostic tools and smart agricultural monitoring systems. However, this study is limited using a publicly available dataset, which may not fully represent real field conditions such as varying lighting, background noise, or occluding leaves. Therefore, future research should focus on testing the model with real-world field images, developing lightweight models for real-time deployment, and integrating decision-support features such as disease treatment or pesticide recommendation systems to enhance practical agricultural applications.

5.2 Recommendation

For future work, the selected model can be deployed in a user-friendly mobile application, enabling farmers to capture leaf images using a smartphone and receive instant disease diagnoses. This real-time, on-site diagnostic tool would reduce dependency on agricultural experts and minimize delays in treatment. In addition, the model could be integrated into autonomous agricultural robots or drones for continuous field surveillance. These robots can scan crops regularly and alert farmers to disease presence without requiring manual inspection, making large-scale monitoring more feasible and efficient. Beyond detection, the system could be extended with a decision-support module that recommends specific pesticide or treatment solutions based on the identified disease class. This would assist farmers in applying the right intervention promptly, reducing the risk of disease spread and lowering unnecessary pesticide usage. By combining deep learning with smart deployment platforms, this project paves the way toward more intelligent, responsive, and sustainable agricultural practices, ultimately contributing to improved crop health, increased yield, and economic benefits for the farming sector.

Acknowledgement

The authors would like to thank Universiti Teknologi MARA (UiTM) for providing the necessary facilities and academic environment that made this work possible. The access to the university's Wi-Fi network, library resources, and lab spaces significantly contributed to the completion of this project.

Conflict of Interest

The authors declare that they have no conflicts of interest. This study did not receive any funding from public, commercial, or not-for-profit agencies.

Author Contribution

The authors confirm contribution to the paper as follows: ***study conception and design:*** *Mazidah Tajjudin, Nurin Jazlina;* ***data collection:*** *Nurin Jazlina;* ***analysis and interpretation of results:*** *Nurin Jazlina, Mazidah Tajjudin;* ***draft manuscript preparation:*** *Nurin Jazlina, Mazidah Tajjudin. All authors reviewed the results and approved the final version of the manuscript.*

References

[1] Ministry of Agriculture and Food Security. (2024). *Malaysia Agrofood in Figures 2023*. Ministry of Agriculture and Food Security.

[2] Saxena, A., Raghuwanshi, R., Gupta, V. K., & Singh, H. B. (2016). Chilli Anthracnose: The Epidemiology and Management. *Frontiers in Microbiology*, *7*. https://doi.org/10.3389/fmicb.2016.01527

[3] Sreya, J., & Arul, L. R. (2021). Machine Learning Techniques In Plant Disease Detection And Classification – A State Of the Art. *INMATEH Agricultural Engineering*, *65*(3), 362–372. https://doi.org/10.35633/inmateh-65-38

[4] Frazier, A. E., & Song, L. (2024). Artificial Intelligence in Landscape Ecology: Recent Advances, Perspectives, and Opportunities. *Current Landscape Ecology Reports*, *10*(1). https://doi.org/10.1007/s40823-024-00103-7

[5] Sarker, I. H. (2021). Deep Learning: a Comprehensive Overview on Techniques, Taxonomy, Applications and Research Directions. *SN Computer Science*, *2*(6). Springer. https://doi.org/10.1007/s42979-021-00815-1

[6] Gupta, J., Pathak, S., & Kumar, G. (2022). Deep Learning (CNN) and Transfer Learning: A Review. *Journal of Physics: Conference Series*, *2273*(1), 012029. https://doi.org/10.1088/1742-6596/2273/1/012029

[7] Mukhamediev, R. I., Popova, Y., Kuchin, Y., & Zaitseva, E. (2022). Review of Artificial Intelligence and Machine Learning Technologies: Classification, Restrictions, Opportunities and Challenges. *Mathematics (2227-7390)*, *10*(15), 2552–2552. Mdpi. https://doi.org/10.3390/math10152552

[8] Kumari, A., Ann Susan Thomas, & J Thasneem. (2023). Examining the research taxonomy of artificial intelligence, deep learning & machine learning in the financial sphere—a bibliometric analysis. *Quality & Quantity*, *58*. https://doi.org/10.1007/s11135-023-01673-0

[9] Mohd Noor, M. H., & Ige, A. O. (2025). A survey on state-of-the-art deep learning applications and challenges. *Engineering Applications of Artificial Intelligence*, *159*, 111225. https://doi.org/10.1016/j.engappai.2025.111225

[10] Yan, Y., Zhang, Y., Wu, X., Guo, Q., Zhang, D., Li, C., Liu, Y., Zhang, J., An, L., Wang, Z., Guo, J., & Chen, L. (2025). Deep learning paradigm shift in composite materials performance prediction: A review of state-of-the-art applications. *Composites Communications*, *58*, 102542. https://doi.org/10.1016/j.coco.2025.102542

[11] Dalal, M., & Mittal, P. (2025). A Systematic Review of Deep Learning-Based Object Detection in Agriculture: Methods, Challenges, and Future Directions. *Computers, Materials & Continua*, *84*(1), 57–91. https://doi.org/10.32604/cmc.2025.066056

[12] Qiu, J., Lu, X., Wang, X., Chen, C., Chen, Y., & Yang, Y. (2024). Research on image recognition of tomato leaf diseases based on improved AlexNet model. *Heliyon*, *10*(13), e33555–e33555. https://doi.org/10.1016/j.heliyon.2024.e33555

[13] Yousaf, J., Zainab Abuowda, Ramadan, S., Salam, N., Eqab Almajali, Hassan, T., Gad, A., Alkhedher, M., & Ghazal, M. (2024). Autonomous smart palm tree harvesting with deep learning-enabled date fruit type and maturity stage classification. *Engineering Applications of Artificial Intelligence*, *139*, 109506–109506. https://doi.org/10.1016/j.engappai.2024.109506

[14] Hukkeri, G. S., Soundarya, B. C., Gururaj, H. L., & Ravi, V. (2024). Classification of Various Plant Leaf Disease Using Pretrained Convolutional Neural Network On Imagenet. *The Open Agriculture Journal*, *18*(1). https://doi.org/10.2174/0118743315305194240408034912

[15] Alam, Md. S., Ali, Md. R., & Rahman, A. (2025). Detection and localization of ripe tomato in greenhouse environment using Keras-based deep learning models. *Journal of Agriculture and Food Research*, *23*, 102182. https://doi.org/10.1016/j.jafr.2025.102182

[16] Kanakam Soujanya, & J. Jabez. (2021). Recognition of Plant Diseases by Leaf Image Classification Based on Improved AlexNet. *2021 2nd International Conference on Smart Electronics and Communication (ICOSEC)*. https://doi.org/10.1109/icosec51865.2021.9591809

[17] Rozlan, S., & Hanafi, M. (2022). Efficacy of chili plant diseases classification using deep learning: a preliminary study. *Indonesian Journal of Electrical Engineering and Computer Science*, *25*(3), 1442. https://doi.org/10.11591/ijeecs.v25.i3.pp1442-1449

[18] Adnan, I. I., Waughfa, M. Z., & Ahmed, T. (2024). Enhancing Chili Plant Disease Detection: A Comprehensive Ensemble Approach Utilizing Pre-Trained Deep Learning Models. *2024 IEEE International Conference on Power, Electrical, Electronics and Industrial Applications (PEEIACON)*, 1–6. https://doi.org/10.1109/peeiacon63629.2024.10800382

[19] Naik, B. N., Malmathanraj, R., & Palanisamy, P. (2022). Detection and classification of chilli leaf disease using a squeeze-and-excitation-based CNN model. *Ecological Informatics*, 101663. https://doi.org/10.1016/j.ecoinf.2022.101663

[20] Adnan, I. I. (2024). *Chili-Dataset,* Retrieved from Kaggle.com: https://www.kaggle.com/datasets/imranulislamadnan/chili-dataset

[21] Hubballi, M. (2023). *chilli_leaf_dataset*. Retrieved from Kaggle.com: https://www.kaggle.com/datasets/mahaninghubballi/chilli-leaf-dataset

[22] M, M. K., Ranjan, A., & Rajendra Machavaram. (2023). In-field Chilli Crop Disease Detection Using YOLOv5 Deep Learning Technique. *2022 IEEE 7th International Conference for Convergence in Technology (I2CT)*, 1–6. https://doi.org/10.1109/i2ct57861.2023.10126468

[23] Ogundokun, R. O., Awoniyi, C., Adebola, N. T., Akinyemi, A., Akinpelu, S. A., & Adigun, M. O. (2025). Hybrid Deep Transfer Learning for Enhanced Brain Tumor Detection through the Integration of MobileNetV2 and InceptionV3. *Procedia Computer Science*, *258*, 2968–2977. https://doi.org/10.1016/j.procs.2025.04.556

[24] Iparraguirre-Villanueva, O., Guevara-Ponce, V., Ofelia Roque Paredes, Sierra-Liñan, F., Zapata-Paulini, J., & Cabanillas-Carbonell, M. (2022). Convolutional Neural Networks with Transfer Learning for Pneumonia Detection. *International Journal of Advanced Computer Science and Applications*, *13*(9). https://doi.org/10.14569/ijacsa.2022.0130963

[25] Li, S., Wang, L., Li, J., & Yao, Y. (2021). Image Classification Algorithm Based on Improved AlexNet. *Journal of Physics: Conference Series*, *1813*(1), 012051. https://doi.org/10.1088/1742-6596/1813/1/012051

[26] Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). ImageNet Classification with Deep Convolutional Neural Networks. *Communications of the ACM*, *60*(6), 84–90. https://doi.org/10.1145/3065386

[27] Olivia Harris, & Michael Andrews. (2024). Effects and results of dropout layer in reducing overfitting with convolutional Neural Networks (CNN). *World Journal of Advanced Engineering Technology and Sciences*, *13*(2), 836–853. https://doi.org/10.30574/wjaets.2024.13.2.0584

[28] Ahmed, M., Afreen, N., Ahmed, M., Sameer, M., & Ahamed, J. (2023). An inception V3 approach for malware classification using machine learning and transfer learning. *International Journal of Intelligent Networks*, *4*, 11–18. https://doi.org/10.1016/j.ijin.2022.11.005

[29] Mujahid, M., Rustam, F., Álvarez, R., Luis Vidal Mazón, J., Díez, I. de la T., & Ashraf, I. (2022). Pneumonia Classification from X-ray Images with Inception-V3 and Convolutional Neural Network. *Diagnostics*, *12*(5), 1280. https://doi.org/10.3390/diagnostics12051280