Pertanika J. Sci. & Technol. 34 (5): 2981 - 3002 (2026)

PERTANIKA JOURNALS

SCIENCE & TECHNOLOGY

Journal homepage: http://www.pertanika.upm.edu.my/

Diseases of Vegetable Plant Recognition Based on Enhanced SqueezeNet

Kifaa Hadi Thanoon[1], Akram A. Abdulrahman[2], Ali Ibrahim Ahmed[3], Omar Ibrahim Alsaif[4]*, and Ibrahim Ahmed Saleh[5]

[1]*Department of Cyber Security Techniques Engineering, Northern Technical University, Mosul 41002, Nineveh, Iraq*
[2]*Department of Computer Science, Alnoor University, Mosul 41002, Nineveh, Iraq*
[3]*Department of Computer Science, College of Education, University of Telafer, Telafer, Nineveh, Iraq*
[4]*Department of Electronic and Communication Technology Engineering, Polytechnic College Mosul, Northern Technical University, Mosul 41002, Nineveh, Iraq*
[5]*Department of Software Engineering, College of Computer Science and Mathematics, University of Mosul, Mosul 41002, Nineveh, Iraq*

ABSTRACT

Agricultural resources significantly affect food security, and plant diseases cause substantial production losses and economic losses. This paper presents a method for detecting plant diseases using the deep learning (DL) algorithm SqueezeNet. To do this, the SqueezeNet network was modified to detect plant diseases. Yet one disadvantage of SqueezeNet is that it has many parameters and achieves quite low accuracy, which limits its use due to its complexity. A new advanced network based on "improved squeeze architecture" was suggested for plant disease detection. This paper, however, presented the network by simplifying its architecture, thereby enhancing its feature-recognition capabilities. Moreover, changes were made to the fire module, including the size of the convolution kernel, the number of network layers, and the number of channels in the Expand layer. Also, the model was combined with the Efficient Channel Attention (ECA) module, which assigns attention values to each channel via local cross-channel interactions, thereby improving the network's ability to learn salient features. The algorithm was compared with three others—LeNet, MobileNet, and traditional SqueezeNet—using the "PlantVillage" dataset. The proposed model achieved a very high accuracy, exceeding the traditional SqueezeNet by 93.8% and reaching a minimum of 84%.

Keywords: Convolutional neural network, ECA module, lightweight, plant disease, SqueezeNet structure

ARTICLE INFO

Article history:
Received: 08 July 2026
Accepted: 28 September 2026
Published: 30 October 2026

DOI: https://doi.org/10.47836/pjst.34.5.18

kh.thanoon@ntu.edu.iq (Kifaa Hadi Thanoon)
akram30@alnoor.edu.iq (Akram A. Abdulrahman)
ali@uotelafer.edu.iq (Ali Ibrahim Ahmed)
Omar.alsaif@ntu.edu.iq (Omar Ibrahim Alsaif)
i.hadedi@uomosul.edu.iq (Ibrahim Ahmed Saleh)
* Corresponding author

e-ISSN: 2231-8526

© Universiti Putra Malaysia Press

INTRODUCTION

In the last few years, global food security has been under severe stress due to different reasons like climate change, the loss of pollinators, and plant diseases, with the latter being the main cause of the drop in the quality and productivity of agricultural products (Guerrero-Ibañez & Reyes-Muñoz, 2023; Panchal et al., 2023; Rahman et al., 2020). When crops are affected by diseases, physiological functions are considerably impaired, leading to stunted growth and poor yields, which in turn negatively impact farmers' economic outcomes. Given that crop plants are typically numerous and small, detecting diseases early is challenging, and by the time they are noticed, the infection may have already spread extensively (Khan et al., 2021; Wang, 2022). Vegetables, which are vital economic crops, are rich in nutrients. Nonetheless, with the expansion of vegetable-planting areas, plant disease incidence has intensified, severely impacting vegetable production (Shoaib et al., 2023; Yang & Liu, 2020). Thereby leading to improved productivity of existing farmland, and ensuring accurate, early plant disease detection and diagnosis is of great importance. Plant diseases pose significant risks to farmers, leading to significant yield losses. The early detection of diseases forms the basis for active diagnosis and control, and the preservation of crop yield and quality. In this paper, plant disease identification technologies are categorised into the following stages:

The human identification approach, which depends on farmers' long-accumulated agricultural knowledge and practical experience, identifies the disease type through visual observation and comparison. Sometimes, farmers could rely on experts for guidance and disease diagnosis, drawing on their extensive inspections, statistics, and measurements (Jaderberg et al., 2014; Zhao et al., 2022). On the other hand, this approach is considered labour-intensive and time-consuming, with several drawbacks, including slow recognition speed, high subjectivity, low accuracy, and an inability to meet the demands of modern agricultural production (Hazarika et al., 2021; Saleh et al., 2021).

The traditional method of machine-learning recognition identifies plant diseases by extracting key features from images using feature engineering, which are then classified by ML classifiers. Usually, this process involves several steps, including plant disease image acquisition, preprocessing, feature extraction, segmentation, and disease class classification. Even though this approach offers considerable benefits over manual methods, such as more consistent results and faster processing, it still has significant limitations, including complex data preprocessing and feature engineering, reliance on manually selected features, relatively low recognition accuracy, and unreliable segmentation. In addition, the identification process still involves a significant amount of human intervention.

Deep learning recognition approach, which can automatically learn complex, "black box" features essential for detection and recognition, thereby eliminating the need for manual feature selection (Mohamed et al., 2022; Oleiwi et al., 2019; Wei & Gao, 2025).

By doing so, it enables an end-to-end system that streamlines the entire process from input to output, providing more accurate and efficient disease identification.

The remaining text of this paper is divided into sections as follows: Section 2 reviews prior research by notable researchers. The method of improving SqueezeNet is discussed in Section 3. Section 4 displays the experimental results, and ultimately, Section 5 includes the conclusion.

LITERATURE REVIEW

In recent years, researchers in the field of plant disease identification have increasingly shifted their focus toward lightweight networks to address the challenges posed by deep networks with numerous layers and complex structures.

In 2023, Jung and other researchers developed a model for detecting crop diseases in the early stages using pairs of diseased and healthy plant images and a CNN consisting of five pre-trained models (Jung et al., 2023). The model followed two stages: crop detection and disease classification. Researchers have achieved a 97.09% accuracy. In 2023, Khan et al. (2023) proposed an application based on MobileNet-V3-small, a DL model for plant disease detection and classification. This study utilised the PlantVillage dataset, which included images of diseased and healthy leaves from 14 crop species and 6 disease groups. The MobileNet-V3-small model has achieved approximately 91.50% test accuracy for leaf classification. This paper has provided a cost-effective solution for farmers to detect diseases early in agricultural applications.

Singh et al. (2022) introduced a model that utilised AI and computer vision for early disease detection in crops such as potatoes, apples, corn, rice, and tomatoes, using the Plant Village dataset (Fu, 2014). The researchers have employed a Bayesian-optimised support vector machine (SVM) for feature extraction and subsequently used a CNN for disease detection. In addition, they have employed a grey-level co-occurrence matrix (GLCM) and histograms of oriented images representing 25 different plants across 58 distinct classes. Multiple network architectures have been trained, with the optimal model achieving a 99.53% performance accuracy. The study concluded with a discussion of the potential of this model as an early-warning tool for plant diseases under real-world cultivation conditions.

It has been demonstrated that combining attention mechanisms with neural network sub-modules can enhance both learning efficiency and overall network performance. However, there are a limited number of studies on the combination of the gradient (HoG) and colour moments. Those 3 feature types — colour, deep features, and texture — have been combined to create hybrid features. A Random Forest classifier has then been utilised to achieve a robust output with minimal features.

Setiawan et al. (2021) presented a paper on the detection of diseases in maize plants using leaf images. They have utilised the PlantVillage dataset, which contains 3852

images across 4 classes: blight, spot, healthy, and rust. This study applied 2 DL networks, SqueezeNet and AlexNet, and compared their performance. Stochastic gradient descent with momentum (SGDM) has been utilised in the optimisation process to improve results. Experimental findings indicate that AlexNet outperforms SqueezeNet in classifying maize leaf diseases.

Mohameth et al. (2020) have utilised 2 ML approaches, applying SVM and KNN algorithms for plant disease classification and feature extraction, and 3 DL models: VGG-16, ResNet-50, and GoogleNet. They have recommended using VGG-16 for image feature extraction and SVM for classification, achieving the highest accuracy of 97.82%. ResNet-50 ranked second with 95.38% accuracy, closely followed by GoogleNet at 95.30%.

Hidayatuloh et al. (2018). Have proposed a study on the early tomato plant disease detection. They have applied CNNs with a variety of architectures, including SqueezeNet, known for its small size. This study is based on a dataset from the Vegetable Crops Research Institute (Balitsa) containing 200 classes. This model could achieve an average identification accuracy of 86.92%.

Ferentinos (2018) developed CNN models for the detection and diagnosis of plant diseases using simple leaf images of healthy and diseased plants. This study utilised a database containing 87,848 integrations of lightweight networks with attention modules. In real-world agricultural production, farmers require solutions for disease identification that can be deployed on mobile devices, necessitating lightweight networks that don't compromise accuracy. This is why the present study improved upon the SqueezeNet model by simplifying its structure and incorporating an attention module. This enhanced model, in addition to occupying less memory, maintains higher accuracy, making it a practical solution for identifying plant diseases. Table 1 summarises related work.

Table 1
Summarised for related works

No.	Author	dataset	What has been done	Accuracy
1	Jung et al., 2023	Plant-Village	Detect and classify plant diseases using a CNN that contains five pre-trained models.	97.09%
2	Khan et al., 2023	Plant-Village	Apply MobileNetV3 to classify plant leaves	91.50
3	Singh et al., 2022	Plant-Village	Classify Diseases of apple, corn, potato, tomato, and rice plants and extract their characteristics using a CNN. Classification using a Bayesian optimised SVM	92.2%
4	Setiawan et al., 2021	Use 3852 images categorised into four groups	Detecting and predicting diseases using e AlexNant and SqueezeNet.	AlexNet = 97.69%, Squeezenet = 44.49%

Table 1 (*continued*)

No.	Author	dataset	What has been done	Accuracy
5	Mohameth et al., 2020	Plant-Village	Apply CNN for feature extraction and SVM and KNN for classification	92.7%
6	(Hidayatuloh et al., 2018)	Plant-Village	Used CNN and SqueezeNet to detect seven types of tomato diseases	86.92%
7	(Ferentinos, 2018)	open database	Apply several models of CNN.	93.8%

MATERIALS AND METHODS

Improved SqueezeNet Neural Network Model

To construct a lightweight, high-precision NN model for plant disease recognition, the classic lightweight network structure SqueezeNet has been modified (Alsaif et al., 2025). In particular, the size of the convolution kernels and the number of channels in the fire module have been adjusted to reduce the network's parameter count. In addition, the Efficient Channel Attention (ECA) module (Hao et al., 2024; Wang et al., 2020) has been incorporated into the network to enhance salient feature extraction and improve recognition efficiency.

Traditional SqueezeNet

SqueezeNet is a lightweight architecture with 50 times fewer parameters than traditional CNNs while maintaining competitive accuracy. The main components of the SqueezeNet architecture are:

Fire Modules : These are the core components of the network, consisting of two layers: the squeeze layer and the expand layer. The squeeze layer applies 1×1 convolution filters to cut down the number of input channels, whereas the expand layer applies a mixture of two filters—one 1×1 convolution and one 3×3 convolution.

Overall Architecture : The architecture begins with a few initial convolutional layers, followed by a series of Fire modules, and concludes with a final convolutional layer, as illustrated in Figure 1.

Reduce Model Size : Use 1×1 convolution filters and downsampling techniques to replace early pooling layers. Pruning techniques are also employed to further reduce the model size.

Batch size : {128, 32, 64}.

Regularization strength : {1e−5, 1e−4, 5e−4}. A value of 1e−4 reduced overfitting without prevention optimisation.

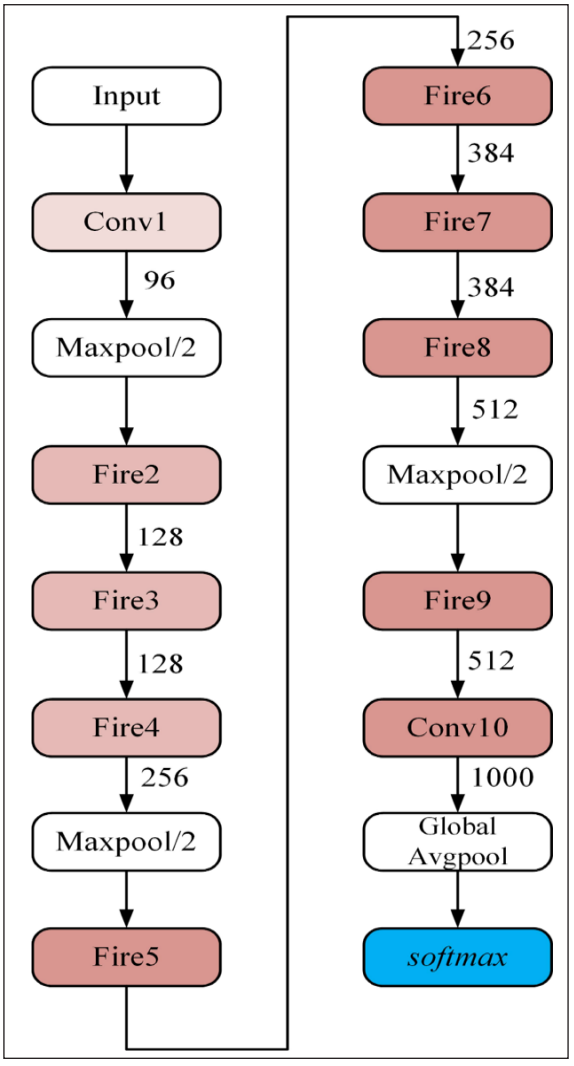

Figure 1. SqueezeNet architecture

Enhanced Fire Module

As the most important component of the SqueezeNet network, the fire module adopts a pioneering approach to reduce both parameter count and overall model size. To further shrink the model, two lightweight strategies to reduce the number of parameters in the network structure. This module divides traditional convolution into a Squeeze layer and an Expand layer. The Squeeze layer utilises 1×1 convolutions, while the Expand layer uses both 1×1 and 3×3 convolutions. This design effectively uses consecutive 1×3 and 3×1 convolutions to replace the original 1×1 and 3×3 convolutions in the fire module, with the number of channels set to half the original value.

The structures are shown in Figures 2 and 3, respectively. (H & W) represent the height and width of the feature map, while (M, S, and E) denote the number of channels. After these improvements, the fire module parameters are given by Equations 1 and 2.

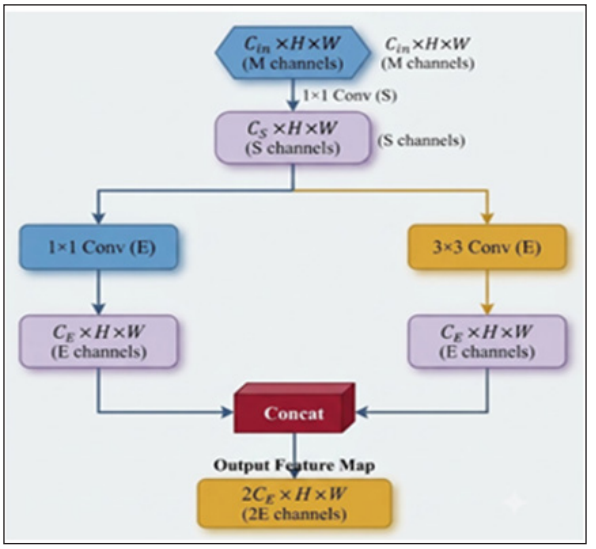

Figure 2. The Fire Module structure

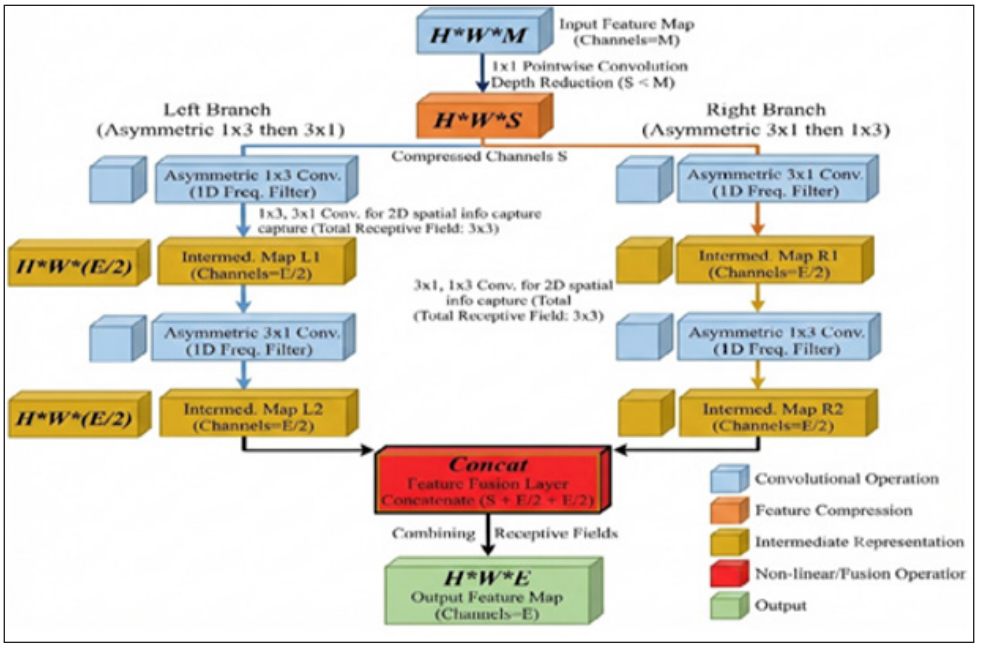

Figure 3. Improved fire module structure

Before the improvement, the parameter quantity of the fire module is

$$N_{before} = S \times 1 \times 1 \times E + S \times 3 \times 3 \times E = 10 \times S \times E \quad [1]$$

The calculated amount is:

$$C_{before} = S \times H \times W \times E \times 1 \times 1 + S \times H \times W \times E \times 3 \times 3 = 10 \times S \times H \times W \times E \quad [2]$$

After improvement, the parameters of the fire module in Equation 3:

$$N_{improve} = 2 \times S \times 1 \times 3 \times \frac{E}{2} + 2 \times \frac{E}{2} \quad 3 \times 1 \times \frac{E}{2} = 3 \times S \times E + \frac{3}{2}E^2 \quad [3]$$

The calculated amount is shown in Equation 4:

$$C_{improve} = 2 \times S \times H \times W \times \frac{E}{2} + 1 \times 3 + 2 \times \frac{E}{2} \times H \times W \times \frac{E}{2} + 3 \times 1 = 3 \times S \times H \times W \times E \times \frac{3}{2} \times H \times W \times E^2 \quad [4]$$

In the SqueezeNet structure, E = 4S. With the improved fire module, the number of parameters and calculations is reduced to 9/10 of the original module, significantly lowering the parameter count in the network. This method effectively reduces the complexity of the classic model.

Efficient Channel Attention (ECA)

Efficient Channel Attention (ECA) is a module based on squeeze-and-excitation blocks, designed to enhance network performance by reducing module complexity without altering dimensionality. To improve the model's ability to learn disease characteristics, the ECA module was added to the SqueezeNet network. This ultra-lightweight attention module assigns distinct attention values to each feature channel, enabling the network to allocate computational resources more efficiently. As a result, the model's parameter count is reduced while accuracy is significantly improved. The ECA module, illustrated in Figure 4, includes the following components:

Global A average pooling : This module compresses global information from the input feature map into a channel descriptor with a global receptive field via the global average pooling layer.

1D convolution : A small kernel size is used to model channel dependencies and capture the correlation between local channels.

Sigmoid function : This function calculates the attention value of each channel based on the correlation between them.

Channel recalibration : This step operates on the input feature map to adjust its channel dimensions. Additionally, the kernel size (K) of the one-dimensional convolution is adaptively determined by the channel dimension, with the calculation formula being as shown in Equation 5.

$$K = \varphi(C) = |\frac{\log_2 C}{r} + \frac{b}{r}|_{odd} \quad [5]$$

Where: C is the number of channels, and "odd" indicates taking the nearest odd number. In this research, r=2, b=1, and K represent the coverage of local cross-channel interaction. Lightweight attention-based networks are designed to realise a balance between computational efficiency and feature representation capability. The networks typically combine compression with attention mechanisms, such as channel or spatial attention, to emphasise informative features while maintaining a low number of parameters. In this context, the proposed method integrates Efficient Channel Attention (ECA) into an enhanced SqueezeNet architecture, aiming to improve disease-feature representation while preserving the lightweight characteristics of the original network.

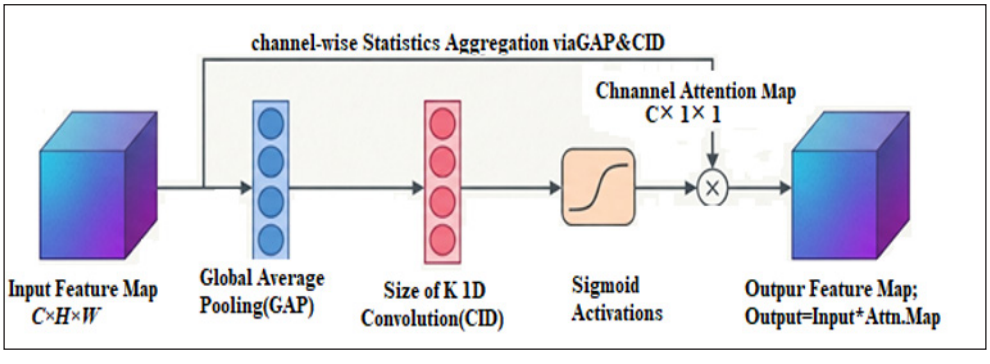

Figure 4. ECA module structure

Enhanced SqueezeNet Network Structure

An enhanced SqueezeNet model is created by combining the improved fire module with the ECA module in three stages, as shown in Figure 5. The first stage involves inputting the disease image from the dataset into the network. Second stage extracts disease features layer by layer through the convolutional layer, pooling layer, and fire module. The final stage processes these extracted feature maps through the ECA module, distinguishing important features from irrelevant ones. The identified important features are used to allocate computing resources efficiently, ultimately achieving higher recognition accuracy with fewer parameters.

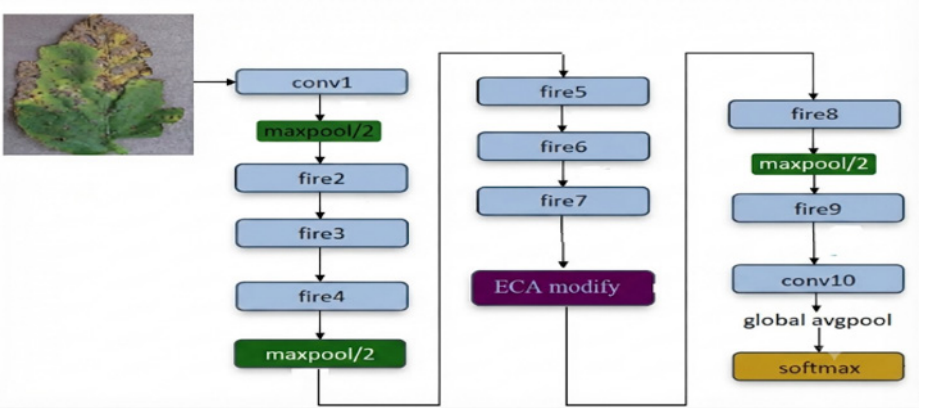

Figure 5. Improved SqueezeNet network architecture

EXPERIMENTAL PARTS

Dataset

This paper relies on the publicly available PlantVillage dataset from Hughes and Salathé (2015) for plant disease images, which includes 54,309 images of healthy and diseased plants across 14 crops. Eight disease categories (including healthy leaves) were selected for identification, with the following distribution: 1,590 healthy leaves, 2,126 bacterial spots, 1,000 early blight, 952 leaf mould, 1,908 late blight, 1,771 spotted blight, 1,676 red spider lesions, and 1,404 leaf spots. Figure 6 illustrates these eight disease types, showing that the leaf backgrounds are simple and free of occlusion.

However, the Plant Village dataset has an uneven category distribution, which can lead to issues in model training, including overfitting. The trained model may be biased toward predicting the category with more samples in the training set, resulting in higher classification errors for minority categories.

To address the data imbalance, data resampling mechanisms are applied to create a more balanced dataset distribution (Fu, 2014). This process helps ensure minimal information

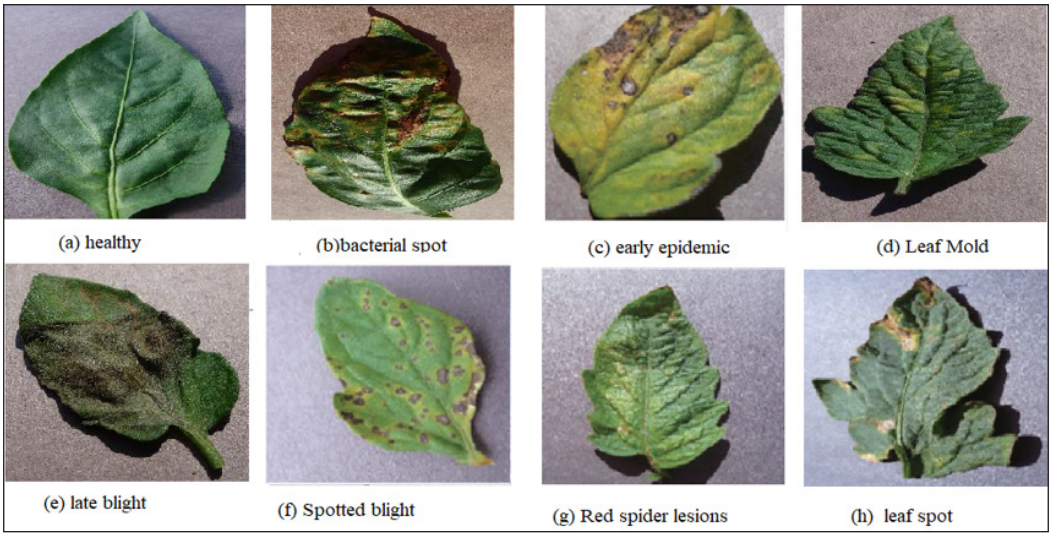

Figure 6. Examples of healthy and diseased plant leaves

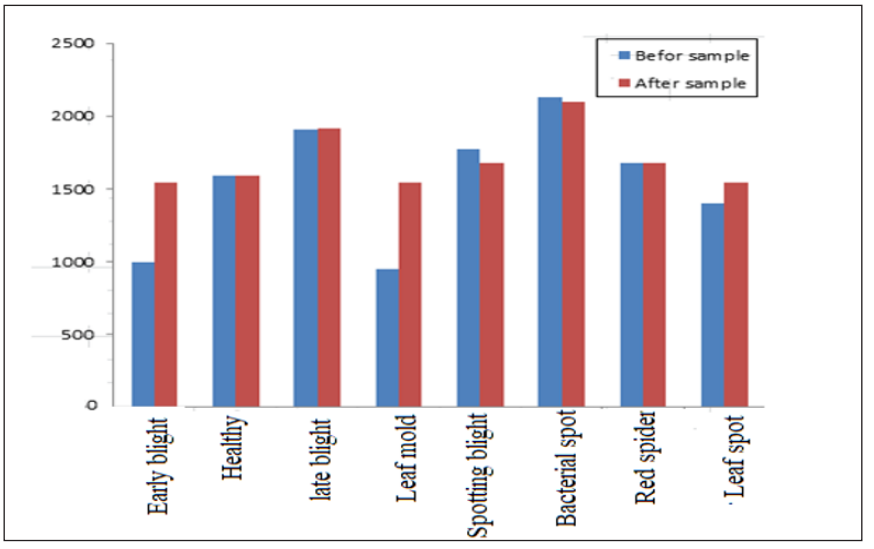

Figure 7. Comparison of data before and after resampling

loss while retaining the most useful data. Specifically, the average number of samples across the eight categories is calculated, and the categories below this average are randomly oversampled by replication, scaling, and other techniques, while the categories above the average remain unchanged. The final dataset includes 13,736 plant disease images. Figure 7 shows the data distribution before and after resampling, with the number of images in each category more balanced, effectively mitigating the sample imbalance between categories.

Model Training Environment Setting

The experiment was conducted on a server with the following hardware configuration: CPU (Intel Xeon E5-2620), 64GB of memory, GPU (NVIDIA GTX 1080 Ti) with 11GB of video memory, and Windows 10. The neural network model was built using the deep learning framework TensorFlow 2.0 within the CUDA environment, in combination with Python 3.6. To comprehensively assess the model's learning performance,

The resampled plant images were divided into training, validation, and test sets in a 7:2:1 ratio, and the neural network was trained accordingly. Based on the total experimental data and empirical values, the number of iterations was set to 100, and the batch sizes for both the training and validation sets were set to 64.

The early stop strategy uses a maximum of 100 training cycles, saves the best checkpoint based on the check loss, and stops training if the loss does not improve for 10 consecutive cycles, thereby ensuring the selection of the model with the best generalisation performance and avoiding over-allocation.

The batch size for the test set was set to 128, and the regularisation coefficient was set to 0.0005. The learning rate and optimiser were selected through experimentation, as outlined in the following section.

Selection of Model Hyperparameters

During training, hyperparameters significantly influence the network's training and recognition performance. Different network structures have different optimal hyperparameters due to varying numbers of layers, parameters, and complexities. To determine the best optimiser for model training, the classic stochastic gradient descent (SGD) method was compared with the Adaptive Moment Estimation (Adam). A systematic hyperparameter search was conducted using a grid search approach over a predefined set of candidate values. The evaluated hyperparameters comprised learning rates of (0.0004), optimisers {SGD, Adam}, batch sizes (64), L2 regularisation coefficients (5×10^{-4}), and early stopping equal 97%. The best trade-off between classification accuracy and training stability was therefore adopted in the remaining experiments. The training process results under each optimiser are shown in Table 2.

Table 2
The training process of the models under the two optimisers

Epoch	Accuracy	
	Adaptive Moment Estimation (Admin)	Stochastic gradient descent (SGD)
20	84.00	18.00
30	86.09	18.98
40	89.12	39.49
50	91.45	56.58
60	92.70	77.45
70	92.82	81.10
80	93.00	85.67
90	96.17	85.32
100	97.29	92.46

For the Adaptive Moment Estimation algorithm (Adam), precision, recall, and f1-score are reported in Table 3.

Table 3
Comparative results for the Adaptive Moment Estimation algorithm improved SqueezeNet

Epoch	Accuracy (%)	Precision (%)	Recall (%)	F-score (%)
20	84.00	88.64	78.00	82.94
30	86.09	89.32	82.00	85.80
40	89.12	91.68	86.00	88.76
50	91.45	92.78	90.00	91.37
60	92.70	93.31	92.00	92.65
70	92.82	92.63	93.00	92.81
80	93.00	92.16	94.00	93.07
90	96.17	97.46%	96.84%	97.15
100	97.29	97.46	96.84	97.15

In the early stages of training, the network's accuracy increased rapidly under the Adam optimiser, with fast convergence and high final accuracy. In contrast, when using the SGD optimiser, the network's accuracy remained unchanged for the first 20 rounds, indicating no improvement. After further learning, accuracy began to increase slowly, starting at 20 rounds. However, even after 100 rounds, the network had not fully converged, and the accuracy remained low. Given the training time and convergence results, Adam was selected as the optimiser for the model. Controlling learning progress is crucial, and hyperparameters such as the learning rate are key to this process.

Tables 2 and 3 show the Adam algorithm as the best optimiser. To increase accuracy, training efficiency metrics were calculated: training time per cycle, actual convergence speed (number of cycles required to reach 95% accuracy), and inference response time, which were used to obtain the final accuracy shown in Table 4.

Table 4
Comparative training performance of optimisers

Optimizer	Final Accuracy (%)	Epochs to 95% Acc.	Training Time/ Epoch (sec)	Totel Training Time(min)	Inference Latency (ms/image)
Admin	97.29	52	42.3	33.8	8.2
SGD	92.46	82	38.7	50.3	7.9

As observed, Adam achieves the best final accuracy (97.29%) with the least training cycles (52) to reach 95% accuracy threshold, demonstrating exceptional convergence speed. Although the SGD algorithm exhibits slightly less training time per cycle (38.7 seconds

vs 42.3 seconds for Adam), its slower convergence rate results in a meaningfully longer total training time,"50.3 minutes vs 33.8 minutes for Adam". This represents a 32.8% reduction in total training time when using Adam. Furthermore, the inference response time remains consistent across all model optimisers (7.9-8.5 ms/image), and the choice of optimiser does not affect the computational cost of the forward-scrolling process after model training. Adam's key advantage lies in its Adaptive Moment Estimation mechanism, which enables faster convergence and more stable training dynamics, as evidenced by the smoother accuracy and loss curves Figure 8. This stability is crucial when training lightweight networks such as the optimised SqueezeNet, where gradient fluctuations can hinder convergence.

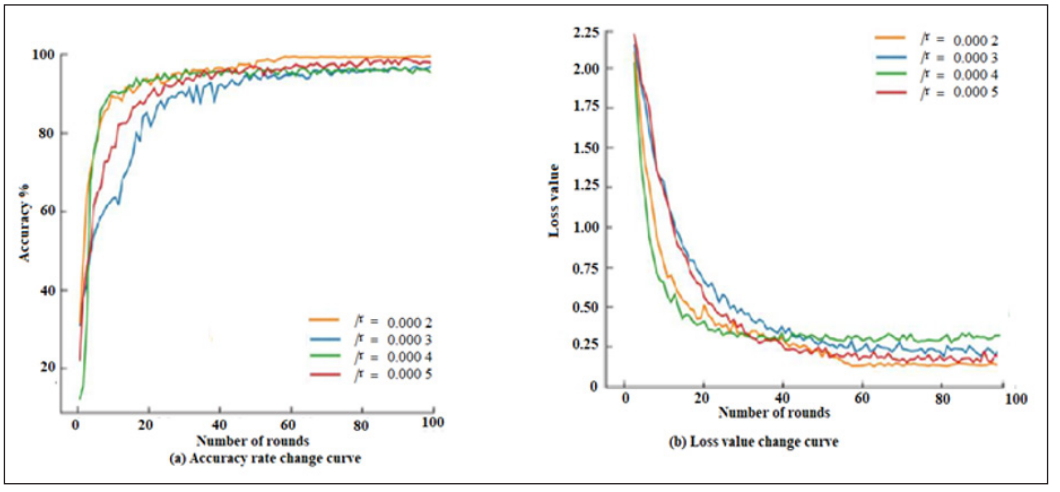

Figure 8. Change the accuracy and loss curves for network training under different learning rates

When the learning rate was set to 0.0002, the network achieved its highest accuracy of 94.70%. However, the accuracy curve showed significant oscillation, indicating that the model was unstable. At a learning rate of 0.0005, the accuracy was 91.76%. However, after 60 training rounds, both accuracy and loss increased, indicating overfitting. At learning rates of 0.0003 and 0.0004, the oscillation in the accuracy curve was minimal, with the accuracy rates reaching 93.61% and 94.07%, respectively. These learning rates allowed for more stable network training. Based on these results, 0.0004 was selected as the optimal learning rate for this paper.

RESULTS AND DISCUSSIONS

The first impact of the improved fire module on model performance. Two network structures were built under the same experimental conditions: one used the traditional SqueezeNet, while the other replaced the fire module in SqueezeNet with the improved fire module,

referred to as Model I. Both models were trained on eight types of plant disease. The results showed that SqueezeNet had 1,255,431 parameters and achieved an accuracy of 95.87%. In comparison, model I had a slightly lower accuracy of 94.07%, but its parameter count was significantly reduced to only 7/10 of SqueezeNet's. This demonstrates that the lightweight strategy applied to the fire module can substantially reduce the model's parameter count and size while incurring only a minor reduction in accuracy.

Second impact of the ECA module on model performance. To analyse the effect of adding the ECA module to the model, the attention module was embedded at four different positions within Model I, and the model was trained on the same processed dataset. The results show that Model I, without the ECA module, had 886,791 parameters and an accuracy of 94.07%. When the ECA module was added after pool 2, fire 7, fire 9, and conv 10, the number of parameters increased slightly to 886,797, with corresponding accuracies of 93.89%, 97.29%, 93.20%, and 95.28%, respectively.

It can be observed that adding the ECA module only slightly increases the number of parameters, but introducing the same attention module at different positions has varying effects on the network's performance. In some cases, it weakens the original recognition ability. In the improved model proposed in this paper, the highest accuracy was achieved when the ECA module was placed after Fire 7. This suggests that placing the ECA module at this position enables the model to effectively identify key features, enhance the expression of useful features, and suppress irrelevant ones, thereby improving the network's accuracy.

To further explore the influence of the ECA module on Model I's stability, Gaussian and salt-and-pepper noise were added to the PlantVillage dataset, and recognition performance was compared with the ECA module embedded at different positions within the network. The stability results, shown in Figure 9, indicate that the model's accuracy decreases

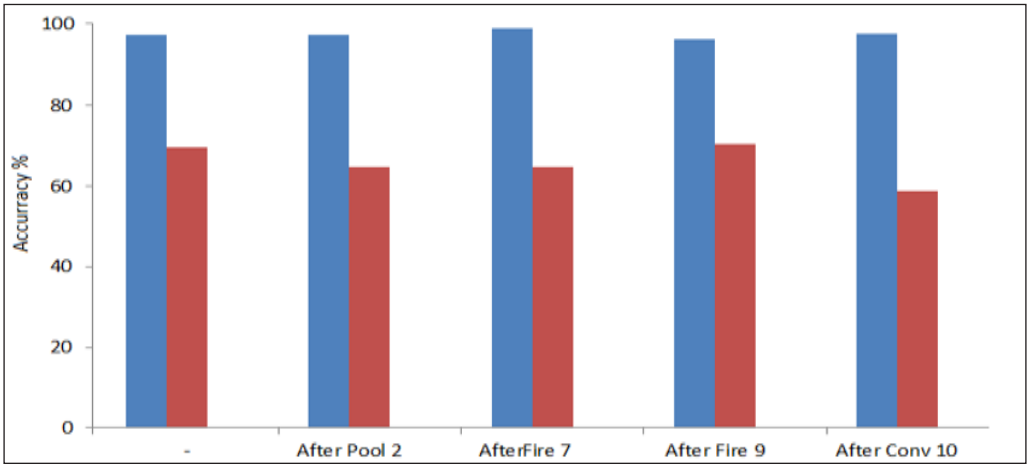

Figure 9. Stability results when the ECA module is at different locations in the network

compared with the noise-free dataset when noise is added. Additionally, network stability improved only when the ECA module was embedded behind fire 9, while it decreased when placed behind pool 2, fire 7, or conv 10. Therefore, placing the ECA module behind Fire 9 not only helps the network learn important disease characteristics but also improves its robustness to noise, leading to better overall stability.

Comprehensive Ablation Study

To evaluate the modifications made to the four network architectures shown in Table 5, the proposed model, which combines two modifications with the improved Fire module, reduces the number of parameters and model size while maintaining competitive accuracy as:

- The ECA module improves feature representation, and combining the two modifications yields the highest discrimination accuracy (97.29%).
- Maintaining the smallest model size, demonstrating that the two components effectively complement each other.

Table 5
Compare parameters between different networks

Model	Accuracy	Parameters	Model Size (MB)
SqueezeNet	95.87	1,255.431	5.0
Fire only	94.07	886791	3.7
ECG only	95,03	1,255,437	5.02
Proposed model	97.29	886,797	3.5

Comparison of improved SqueezeNet and other models. Table 6 presents performance comparisons of different network models on plant disease images. AUC for all those algorithms with the confusion matrix of the proposal method shown in Figure 10.

Table 6
Performance comparison of different network models

Model	Accuracy %	Precision %	Recall %	F-score %	Model size / MB
LeNet	89.52	89.11	90.00	0.8955	28.64
MobileNet	91.19	90.87	91.60	91.23	13.00
SqueezeNet	95.87	95.91	95.80	95.85	5.00
Improved SqueezeNet	97.29	97.35	97.22	97.28	3.50
CNN	97.08	97.14	97.01	97.07	32.78
CNN and SqueezeNet	86.92	87.10	86.75	86.92	-

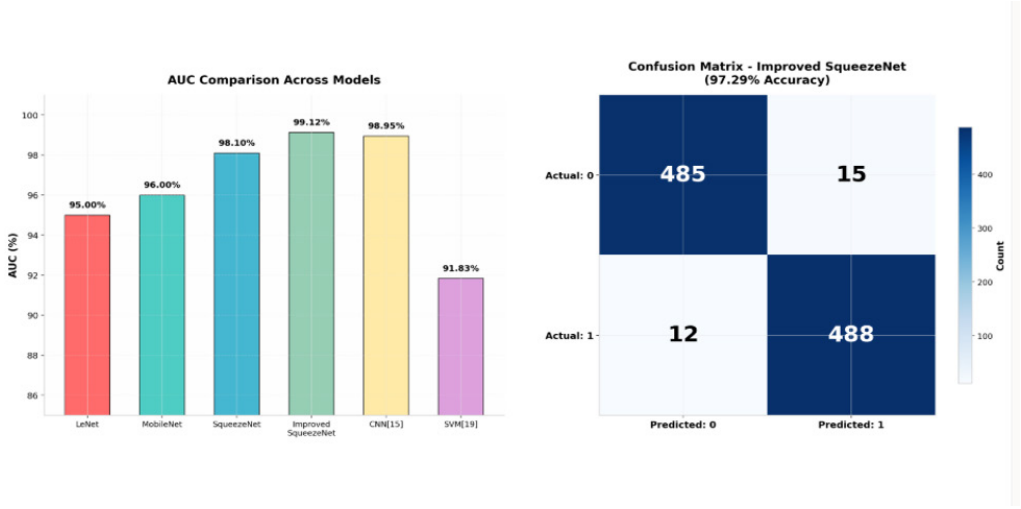

Figure 10. AUC curve and confusion matrix

As seen in Table 6, compared to the classic network model LeNet, the improved SqueezeNet model demonstrates clear advantages in both accuracy and model size. Compared with other lightweight network models, such as MobileNet and traditional SqueezeNet, the improved SqueezeNet achieves higher average accuracy while requiring less memory. Therefore, the improved SqueezeNet model outperforms the three existing network models in terms of both accuracy and size, making it well-suited for applications requiring high accuracy and a lightweight design.

Feature Visualisation and Attention Analysis

This section explains the mechanisms for extracting and enhancing attention to detail. To evaluate the proposed architectural modifications, a qualitative analysis was conducted by visualising the model's internal feature representations.

The feature maps from the last convolutional layer (conv10) of both conventional and enhanced SqueezeNet models. As shown in Figure 11, the feature maps from the enhanced model exhibit more distinct, focused activations, primarily concentrating on diseased areas of the leaf, such as spots and lesions. In contrast, the conventional SqueezeNet shows more dispersed and less distinct activation patterns. This improvement is attributed to modifications in the Fire module. The use of asymmetric convolutions (1×3 and 3×1) in the extended layers expands the receiving area and encourages the network to learn a more diverse set of features, enabling the effective capture of elongated and irregular shapes that characterise plant disease symptoms.

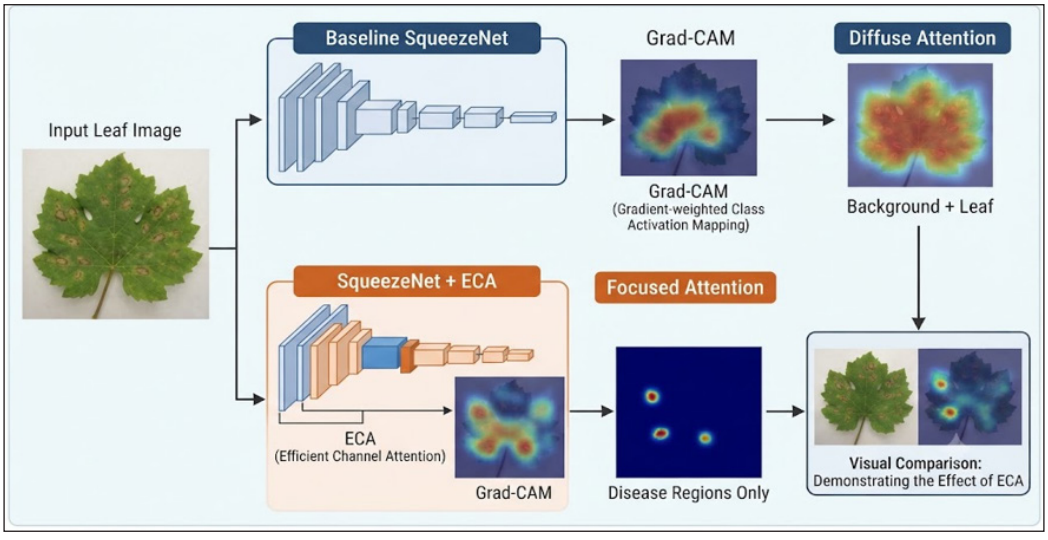

Figure 11. Feature maps from the final convolutional layer for SqueezeNet

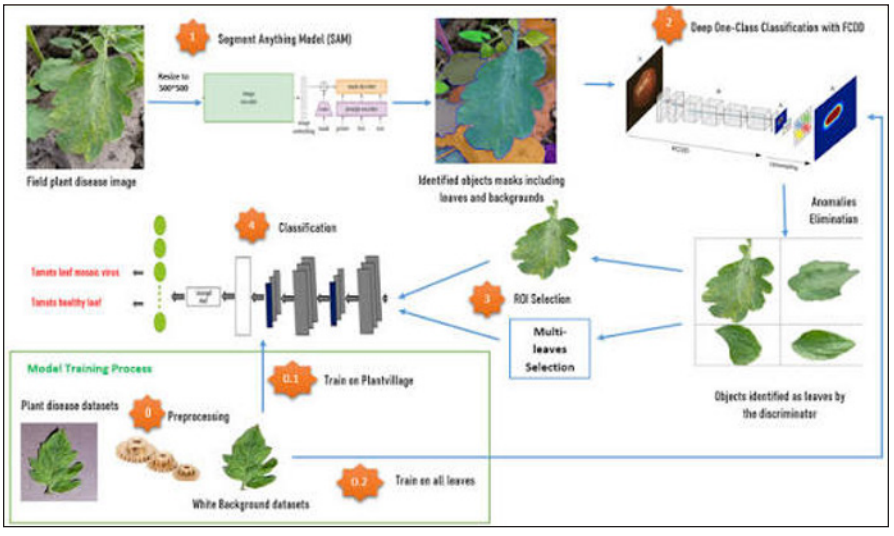

Figure 12. The ECA module

To interpret the role of the ECA module, it employed Gradient-weighted Class Activation Mapping (Grad-CAM) to generate attention heatmaps. As illustrated in Figure 12, the heatmaps reveal the regions of the input image that are most influential in the model's classification decision. The baseline SqueezeNet model often focuses on both the diseased areas and the healthy background or leaf edges, indicating a lack of precise attention to disease-specific features. However, when the ECA module is integrated, the network's attention becomes highly localised and specific to the diseased spots. This is because the

ECA module assigns adaptive weights to each feature channel, effectively amplifying channels that contain crucial information about the disease (e.g., colour and texture anomalies) while suppressing irrelevant channels. The local cross-channel interaction captured by the 1D convolution in the ECA module is key to this precise focusing ability, enabling feature recalibration without significant computational overhead. Table 7 briefly explains the methodology; it requires more detail, including a complete layer-by-layer network configuration.

Table 7
Complete layer-by-layer network configuration

Layer	Type	Kernel Size	Stride Input	Channels	Output Channels	Activation function	Parameters
Conv1	Convolution	7×7	2	3	96	ReLU	14,112
Pool1	Max Pooling	3×3	2	96	96	-	0
Fire2	Fire Module (improved)	-	-	96	128	ReLU	12,544
Pool2	Max Pooling	3×3	2	128	128	-	0
Fire3	Fire Module (improved)	-	-	128	192	ReLU	| 19,584
Fire4	Fire Module (improved)	-	-	192	256	ReLU	27,264
Pool4	Max Pooling	3×3	2	256	256	-	0
Fire5	Fire Module (improved)	-	-	256	384	ReLU	54,528
Fire6	Fire Module (improved)	-	-	384	512	ReLU	91,392
ECA	Attention Module	(1D Conv k=5)	-	512	512	Sigmoid	| 2,560
Fire7	Fire Module (improved)	-	-	512	512	ReLU	121,344
Fire8	Fire Module (improved)	-	-	512	768	ReLU	182,016
Fire9	| Fire Module (improved)	-	-	768	1024	ReLU	324,864
Conv10	Convolution	1×1	1	1024	8	ReLU	8,200
Pool10	Global Average Pooling	8×8	1	8	8	-	0
Softmax	Classification	-	-	8	8	Softmax	0

CONCLUSION

This paper applies deep learning to plant disease detection by proposing an enhanced SqueezeNet model incorporating an ECA module. The model includes a lightweight fire module and an attention mechanism. The fire module uses asymmetric convolutions

(1×3 and 3×1) and channel reduction, reducing the number of parameters and model size: from 1,255,431 parameters and 5.0 MB in the traditional SqueezeNet to 886,797 parameters and 3.5 MB in the improved version.

Furthermore, the ECA module enabled the network to adaptively emphasise the most discriminative disease-related features, thereby accurately localising disease symptoms such as spots and lesions. Experimental results demonstrate that the proposed model achieves higher classification accuracy than several existing DL models while maintaining the smallest memory footprint, making it suitable for resource-constrained and portable applications. Moreover, the ablation study confirms that combining the improved Fire module and the ECA module yields complementary benefits, achieving the best trade-off between accuracy and model efficiency. Since experiments were conducted on a public dataset that contains a single leaf and a single disease per image, further validation under real agricultural conditions is still required.

In future work, a set of real field images will be used. Preprocessing will be applied to them, using geometric transformations to address lighting and colour disturbances, such as adjusting lighting and contrast, image size, and rotation, which help reduce the contrast between data distributions.

REFERENCES

Alsaif, O. I., Hasan, S. Q., & Maray, A. H. (2023). Using skeleton model to recognize human gait gender. *IAES International Journal of Artificial Intelligence (IJ-AI), 12*(2), 974-983. https://doi.org/10.11591/ijai.v12.i2.pp974-983

Chappa, R. T. N. V. S., & El-Sharkawy, M. (2020). Squeeze-and-excitation SqueezeNext: An efficient DNN for hardware deployment. In *2020 10th Annual Computing and Communication Workshop and Conference (CCWC)* (pp. 691-697). IEEE. https://doi.org/10.1109/ccwc47524.2020.9031119

Ferentinos, K. P. (2018). Deep learning models for plant disease detection and diagnosis. *Computers and Electronics in Agriculture, 145*, 311-318. https://doi.org/10.1016/j.compag.2018.01.009

Fu, P. (2014). *Research on classification methods for large-scale imbalanced data* [Thesis, Liaoning Normal University].

Guerrero-Ibañez, A., & Reyes-Muñoz, A. (2023). Monitoring tomato leaf disease through convolutional neural networks. *Electronics, 12*(1), Article 229. https://doi.org/10.3390/electronics12010229

Hao, M., Sun, Q., Xuan, C., Zhang, X., & Zhao, M. (2024). SqueezeNet: An improved lightweight neural network for sheep facial recognition. *Applied Sciences, 14*(4), Article 1399. https://doi.org/10.3390/app14041399

Hazarika, R. A., Abraham, A., Kandar, D., & Maji, A. K. (2021). An improved LeNet-deep neural network model for Alzheimer's disease classification using brain magnetic resonance images. *IEEE Access, 9*, 161194-161207. https://doi.org/10.1109/access.2021.3131741

Hidayatuloh, A., Nursalman, M., & Nugraha, E. (2018). Identification of tomato plant diseases by leaf image using SqueezeNet model. In *2018 International Conference on Information Technology Systems and Innovation (ICITSI)* (pp. 199-204). IEEE. https://doi.org/10.1109/ICITSI.2018.8696087

Hughes, D. P., & Salathé, M. (2015). *An open access repository of images on plant health to enable the development of mobile disease diagnostics* [Preprint]. arXiv. https://doi.org/10.48550/arXiv.1511.08060

Iandola, F. N., Han, S., Moskewicz, M. W., Ashraf, K., Dally, W. J., & Keutzer, K. (2016). *SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and <0.5MB model size* [Preprint]. arXiv. https://doi.org/10.48550/arXiv.1602.07360

Jung, M., Song, J. S., Shin, A.-Y., Choi, B., Go, S., Kwon, S.-Y., Park, J., Park, S. G., & Kim, Y.-M. (2023). Construction of deep learning-based disease detection model in plants. *Scientific Reports, 13*(1), Article 7331. https://doi.org/10.1038/s41598-023-34549-2

Khan, A. T., Jensen, S. M., Khan, A. R., & Li, S. (2023). Plant disease detection model for edge computing devices. *Frontiers in Plant Science, 14*, Article 1308528. https://doi.org/10.3389/fpls.2023.1308528

Khan, R. U., Khan, K., Albattah, W., & Qamar, A. M. (2021). Image-based detection of plant diseases: From classical machine learning to deep learning journey. *Wireless Communications and Mobile Computing, 2021*(1), Article 5541859. https://doi.org/10.1155/2021/5541859

Lahgazi, M. J., & Mohaoui, S. (2025). *Attfire: Attention-enhanced fire modules for image segmentation* [Preprint]. SSRN. https://doi.org/10.2139/ssrn.5356529

Mohamed, S. A., Alsaif, O. I., & Saleh, I. A. (2022). Intrusion detection network attacks based on whale optimization algorithm. *Ingénierie des Systèmes d'Information, 27*(3), 441-446. https://doi.org/10.18280/isi.270310

Mohameth, F., Bingcai, C., & Sada, K. A. (2020). Plant disease detection with deep learning and feature extraction using Plant Village. *Journal of Computer and Communications, 8*(6), 10–22. https://doi.org/10.4236/jcc.2020.86002

Oleiwi, Z., Thanoon, K., & Alsaif, K. (2019). High frequency coefficient effect on image based on contourlet transformation. In *2019 International Conference on Computing and Information Science and Technology and Their Applications (ICCISTA)* (pp. 1-4). IEEE. https://doi.org/10.1109/iccista.2019.8830649

Panchal, A. V., Patel, S. C., Bagyalakshmi, K., Kumar, P., Khan, I. R., & Soni, M. (2023). Image-based plant diseases detection using deep learning. *Materials Today: Proceedings, 80*, 3500-3506. https://doi.org/10.1016/j.matpr.2021.07.281

Rahman, C. R., Arko, P. S., Ali, M. E., Iqbal Khan, M. A., Apon, S. H., Nowrin, F., & Wasif, A. (2020). Identification and recognition of rice diseases and pests using convolutional neural networks. *Biosystems Engineering, 194*, 112-120. https://doi.org/10.1016/j.biosystemseng.2020.03.020

Saleh, I. A., Albayati, A. H., & Thanoon, K. H. (2021). Measure the software quality based on grasshopper optimization algorithm. *International Journal of Computing and Digital Systems, 10*(1), 955-961. https://doi.org/10.12785/ijcds/100186

Setiawan, W., Ghofur, A., Hastarita Rachman, F., & Rulaningtyas, R. (2021). Deep convolutional neural network AlexNet and SqueezeNet for maize leaf diseases image classification. *Kinetik: Game Technology,*

Information System, Computer Network, Computing, Electronics, and Control, 6(4). https://doi.org/10.22219/kinetik.v6i4.1335

Shoaib, M., Shah, B., El-Sappagh, S., Ali, A., Ullah, A., Alenezi, F., Gechev, T., Hussain, T., & Ali, F. (2023). An advanced deep learning models-based plant disease detection: A review of recent research. *Frontiers in Plant Science, 14*, Article 1158933. https://doi.org/10.3389/fpls.2023.1158933

Singh, A. K., Sreenivasu, S. V. N., Mahalaxmi, U. S. B. K., Sharma, H., Patil, D. D., & Asenso, E. (2022). Hybrid feature-based disease detection in plant leaf using convolutional neural network, Bayesian optimized SVM, and random forest classifier. *Journal of Food Quality, 2022*(1), Article 2845320. https://doi.org/10.1155/2022/2845320

Wang, B. (2022). Identification of crop diseases and insect pests based on deep learning. *Scientific Programming, 2022*, Article 9179998. https://doi.org/10.1155/2022/9179998

Wang, Q., Wu, B., Zhu, P., Li, P., Zuo, W., & Hu, Q. (2020). ECA-Net: Efficient channel attention for deep convolutional neural networks. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 11531-11539). IEEE. https://doi.org/10.1109/cvpr42600.2020.01155

Wei, K., & Gao, Y. (2025). Crop disease and pest identification based on deep learning and Raspberry Pi. In *Proceedings of the 2025 7th Asia Conference on Machine Learning and Computing* (pp. 47-53). Association for Computing Machinery. https://doi.org/10.1145/3772673.3772703

Yang, T., & Liu, C. (2020). Recognition system for leaf diseases of Ophiopogon japonicus based on PCA-SVM. *Plant Diseases and Pests, 11*(2), 11-15.

Zhao, S., Liu, J., Bai, Z., Hu, C., & Jin, Y. (2022). Crop pest recognition in real agricultural environment using convolutional neural networks by a parallel attention mechanism. *Frontiers in Plant Science, 13*, Article 839572. https://doi.org/10.3389/fpls.2022.839572