Integrating Squeeze-and-Excitation Network with Pretrained CNN Models for Accurate Plant Disease Detection

Original Scientific Paper

Lafta Raheem Ali
General Directorate of Education of Salahuddin
Salahuddin, Iraq
l.alkhazraji@gmail.com

Sabah Abdulazeez Jebur*
Imam Alkadhim University College,
Department of Cyber Security
Baghdad ,Iraq
sabah.abdulazeez@iku.edu.iq

Mothefer Majeed Jahefer
Imam Alkadhim University College,
Department of Computer Science
Baghdad ,Iraq
modafarmajed@iku.edu.iq

*Corresponding author

Abbas Khalifa Nawar
Imam Alkadhim University College, Department of Computer Science
Baghdad ,Iraq
abbas.altimimy@iku.edu.iq

Zaed S. Mahdi
University of Technology,
Information Technology Center
Baghdad ,Iraq
zaed.s.mahdi@uotechnology.edu.iq

Abstract – *The increasing global population and the challenges posed by climate change have intensified the demand for sustainable food production. Traditional agricultural practices are often insufficient, leading to significant crop losses due to diseases and pests, despite the widespread use of pesticides and other chemical interventions. This paper introduces a new approach that integrates deep learning techniques, specifically Convolutional Neural Networks (CNNs) with Squeeze and Excitation (SE) networks, to enhance the accuracy of disease detection in fig leaves. By leveraging three pre-trained CNN models—MobileNetV2, InceptionV3, and Xception—this framework addresses data scarcity issues and improves feature representation while minimizing the risk of overfitting. Data augmentation techniques were employed to counteract data imbalance, and visualization tools like Grad-CAM and t-SNE were utilized for model interpretability. The proposed CNN-SE model was trained and evaluated on a fig leaf dataset comprising 1,196 images of healthy and diseased fig leaves, achieving an accuracy of 92.90% with MobileNet-SE, 91.48% with Inception-SE, and 89.62% with Xception-SE. Our model demonstrates superior performance in detecting fig leaf diseases, presenting a robust solution for sustainable agriculture by providing accurate, efficient, and scalable disease management in crops. The code of the proposed framework is available at https://github.com/lafta/SE-block-with-CNN-Models-for-Plant-Disease-Detection.*

Keywords: *Deep Learning , Convolutional Neural Network ,Squeeze-and-Excitation , Plants diseases detection*

Received: January 25, 2025; Received in revised form: May 3, 2025; Accepted: May 30, 2025

1. INTRODUCTION

One of the greatest challenges to increasing agricultural productivity is the spread of pests and diseases in crops, which are considered the primary cause of more than a third of annual agricultural production losses [1]. To protect plants from these threats, numerous pesticides and costly techniques are employed. However, the large-scale use of these chemical methods has adverse effects on species diversity, human health, and crop yields, while also increasing production costs [2]. Recently, researchers have reported remarkable progress in applying Artificial Intelligence (AI) technology, particularly deep learning (DL) techniques, to the detection and classification of diseases on plants. These techniques have played a key role in transforming conventional farming practices into more sustainable ones by providing accurate, efficient, and scalable solutions, even aiding in the early diagnosis of diseases [3]. DL algorithms can classify image data based on their feature content and extract relevant information [4]. Convolutional Neural Networks (CNNs) are a specialized type of DL models primarily designed to process image data, automatically learning features and patterns before making decisions [5]. The extracted features are fed into the classifier without human intervention, unlike in traditional machine learning, where feature extraction and classification are separate steps. Essentially, CNNs are composed of two steps; feature extraction and classification, the first step are employed three operations which are convolution operation that achieved by convolutional layer, activation function, and pooling operation. Various filters are applied to analyze and detect the important features in the image starting from small features such as edges, lines, and corners till reach the very important features (faces, leaves, etc.) [6]. From the perspective of feature re-calibration, a Squeeze and Excitation (SE) network has been introduced to capture the interdependencies between convolutional feature channels [7]. The SE block consists of two main processes: squeezing and excitation. The squeeze operation creates a channel descriptor by summarizing feature maps across their spatial dimensions to embed global information. The excitation process generates channel-specific weights. Through feature re-calibration, the SE block can selectively highlight important features while diminishing less relevant ones. This block can be incorporated into conventional DL models, such as CNNs [8]. Despite all the capabilities CNNs offer, they still face several challenges, the most significant being the need for large amounts of training data [9]. This has prompted researchers in the field of AI to explore the use of transfer learning (TL), a technique that improves model performance by transferring knowledge from an already trained model instead of training the model from scratch [10]. In addition to TL, data augmentation is a technique used to handle the lack of data by increasing the size of the training dataset through various transformations of existing images, such as translation, rotation, shearing, flipping, zooming, etc. This generates new data that is added to the original dataset, enhancing the model's generalization and robustness [11]. Since DL as a black box, it is difficult to understand what occurs within the hidden layers and how these networks make decisions or predictions [12]. To address this challenge, transparency is necessary to identify the regions the model focuses on. This can be achieved using explainable learning techniques. Specifically, Grad-CAM and t-SNE visualization techniques are employed to bridge this gap, providing deeper insight and a clearer understanding of how the model reaches its conclusions. This study aims to address the aforementioned challenges by developing an accurate plant disease detection model through feature extraction from leaf images using multiple CNN architectures, and by integrating a squeeze-and-excitation (SE) block to enhance classification accuracy. The main contributions of this paper include:

- A new CNN-SE framework integrating CNNs with SE network has been proposed. This approach enhances feature learning by focusing on informative channels and dynamically recalibrating weights, improving fig leaf disease detection accuracy.
- Three pre-trained CNN models from ImageNet were employed to mitigate data scarcity, improve feature extraction, and reduce overfitting risks.
- Data augmentation techniques were applied to address class imbalance and limited training data, enhancing model generalization.
- Interpretability tools, Grad-CAM and t-SNE, were used to analyze model decisions, visualize feature importance, and detect potential biases.

2. LITERATURE REVIEW

This section presents the most recently studies in the field of detecting the plant leaf disease using the DL techniques. Saikat Datta and Nitin Gupta [13] developed a deep CNN-based architecture to classify tea leaf diseases into six categories: Gray Blight, Algal Spot, Brown Blight, Heliopolis, Healthy Leaves, and Red Spot. They introduced a novel real-world dataset containing 5,867 images, covering five disease types and healthy leaves. F. Khan et al. [14] proposed a DL framework for detecting blight, leaf spot and sugarcane mosaic virus in maize crops. they evaluated five YOLO variants (YOLOv3-tiny, YOLOv4, YOLOv5s, YOLOv7s and YOLOv8n) and selected YOLOv8n for its compact architecture and superior inference speed. this model was subsequently deployed in a mobile application to real-time disease management in agricultural settings. MKA Mazumder et al. [15] proposed LeafDoc-Net, lightweight TL architecture that integrates two pretrained CNN models, DenseNet-121 and MobileNetV2, for multi-species leaf disease detection. The model employs an attention-based transition mechanism for enhancing feature fusion, followed by global average pooling to reduce spatial dimensionality. Additionally, it incorporate dense layers with swish activation and batch normalization to deepen the network while maintaining computational efficiency. Qinghai Wu et al. [16] proposed DL model contains three components of feature extraction, attention calculation and then lastly the classification, an attention module was added to generate feature maps at various depths for enhancing the network's focus on discriminative features while reduce background noise. The attention module also made use of LeakyReLU as an activation function to tackle the problem of neurons failing to learn when their input is negative, The extracted features were integrated through a fully connected layer to predict disease category for soybean leaf. YA Bezabh et al. [17] proposed a pepper disease classification model based on two CNN architectures: AlexNet and VGG16. The authors utilized these two CNN architectures to extract features, then combined the extracted features in single features set.

After that the combined feature set was used as input to the fully connected layers for classification with a multiclass classifier. Rina Bora et al. [18] developed a framework known as the Multivariate Normal Deep Learning Neural Network (MNDLNN) to detect diseases in the leaves, fruits, roots and stems of tomato plants. The methodology comprises of conversion the image color to HSI format, masking of green color to obtain the healthy and unhealthy region, identification of fruits and roots with the region of interest, segmenting the unhealthy region via RKMC clustering and final stage includes the extraction of necessary features using RMSSO. Anuradha Chug et al. [19] proposed a Hybrid Deep Learning (HDL) framework that combines EfficientNet architectures (B0–B7) as feature extractors with five machine learning classifiers. They developed the IARI-TomEBD dataset, a real-time image collection of tomato early blight disease for experimental validation. The HDL models demonstrated strong performance on this custom dataset and were further evaluated on two public plant disease benchmarks. The EfficientNet-B3-ADB and EfficientNet-B3-SGB configurations achieved state-of-the-art results across all datasets. Mahum, Rabbia, et al. [20] proposed an Enhanced DenseNet model by integrating an additional transition layer into DenseNet-201. To address extreme class imbalance in the training data, they employed a reweighted cross-entropy loss function, enhancing model robustness. Ashwathnarayan Nagarjun et al. [21] proposed a cotton leaf disease classification method combining transfer learning and deep learning techniques. For the deep learning component, they employed a conventional convolutional neural network (CNN), while their transfer learning approach utilized architectures such as Inception and ResNet. The study relied on a custom-collected cotton disease dataset to achieve its objectives. However, the work exhibits significant ambiguities and lacks critical implementation details, including methodological transparency and reproducibility safeguards. Malathi Chilakalapudi and Sheela Jayachandran [22] proposed a framework that employs transfer learning-based CNN and a Chronological Flamingo Search Algorithm (CFSA). The authors utilized the color PlantVillage dataset and applied an augmentation process incorporating operations such as contrast adjustment, rotation, rescaling, and others. Manjunatha Shettigere Krishna et al. [23] developed a classification system for detecting plant diseases in leaves using multiple CNN architectures. Their primary contribution involved enhancing data augmentation by introducing Gaussian noise. The authors implemented four CNN architectures in parallel and evaluated their performance across two datasets. In their baseline approach, they processed input data directly through the CNNs without additional modifications, yielding preliminary results. Sherihan Aboelenin et al. [24] developed a method that employs multiple CNN variants and a Vision Transformer (ViT), merging them into an ensemble model. Both CNNs and the ViT were used to extract features: the CNN variants captured global features, while the ViT focused on extracting local features. The model was trained using the Apple and Corn leaf disease datasets from PlantVillage. The global features extracted by the CNN variants were concatenated and fed into the ViT, where they were combined with the local features. The ViT then performed the final classification of leaf diseases. The key contribution of this study lies in the novel integration of CNN architectures with a ViT framework. Table 1 presents the methods, limitations, datasets, and accuracy metrics of the respective studies.

3. METHODS AND MATERIALS

3.1. DATASET.DESCRIPTION

The Fig Leaves Dataset [25] was employed in this work to train and evaluate the proposed model. It comprises 2,321 high-resolution images of fig leaves from various regions in Iraq, captured during the peak fruit season to ensure the utmost accuracy in identifying infections. These images are divided into two categories: infected and healthy leaves. The dataset is both small and unbalanced, with 1,350 images of infected leaves and 971 images of healthy leaves. To tackle the challenges of data imbalance and scarcity, a data augmentation method was implemented in two phases. The first phase involved randomly selecting and duplicating healthy leaf images until their number matched that of the infected leaf images, ensuring equal representation of both classes. In the second phase, standard data augmentation techniques were applied, including rotation, width and height shifts, shear and zoom transformations, horizontal flipping, and rescaling. These techniques were used to increase the training data, enhancing its diversity and robustness. After augmentation, the dataset was randomly split into 80% of the images per class for training and 20% for testing. Fig. 1 displays samples from the fig leaves dataset.

Fig. 1. Samples from the fig leaves dataset. The first row depicts healthy leaves, while the second row depicts infected leaves

3.2. CNN ARCHITECTURES

The application of DL algorithms improves the diagnosis process of plant diseases. Such algorithms work best when analyzing large image databases along with access to strong computational availability [26]. These models coordinate all modelling procedures which start from data pre-processing and move through architecture engineering until they reach hyperparameter optimization and parameter selection or update [27]. The paper investigates leaves infected plant identification by utilizing three deep CNN models comprising MobileNetV2, InceptionV3 and Xception. Testing confirmed these models function well on the ImageNet dataset and extract fine and large features because they contain distinctive filter sizes from 1×1 to 7×7. These models adopt batch normalization layers to speed up learning processes while offering better efficiency in plant disease detection. The mobile-oriented model MobileNetV2 functions as a compact yet efficient system for embedded devices with its design combining 19 bottleneck residual layers and ReLU activation [28]. The structured framework of InceptionV3 comprises three divisions including a stem section along with inception blocks along with final layers which enables the extraction of features from multiple scales and performs classification through a combination of GAP and fully connected layers [29]. Xception uses depthwise separable convolutions to process information faster while decreasing parameter numbers through depthwise and pointwise convolution operations [30].

Table 1. Summary of Related Works: Methods, Datasets, Limitations, and Performance in Leaf Disease Classification

[Ref.], year	Method	Limitations	Dataset	Accuracy
[13], 2023	Deep CNN	The study faces limitations of class imbalance in the dataset and high computational requirements during model training	Tea leaf diseases dataset	96.56%
[14], 2023	YOLOv8n	Use test datasets with uneven class distributions, skewing accuracy metrics and reducing real-world applicability, also, relies on corn leaf images captured with a limited-range camera, introducing device-specific biases. Additionaly, remains non-public, hindering reproducibility.	Corn leaf dataset	99.04%
[15], 2023	LeafDoc-Net	small dataset size, with some classes containing only 39 images. This constraint hinders model generalization, exacerbating overfitting and class imbalance.	corn disease dataset and a wheat leaf sickness dataset	99%
[16] , 2023	CNN	The model exhibits high computational complexity and ignores data balancing in both original and augmented datasets.	Soybean leaf disease dataset	85.42%
[17] , 2023	AlexNet and VGG16	The study relies on conventional CNN architectures, lacks advanced techniques such as attention mechanisms, and involves computationally intensive implementations due to the large number of parameters.	Pepper leaf disease	95.82%
[18] , 2023	Multivariate Normal Deep Learning Neural Network.	The dataset is inaccessible, and the authors omit testing on universal benchmarks like PlantVillage	private dataset	99.84%
[19] , 2023	EfficientNet-B3- ADB and EfficientNet-B3-SGB	Persistent class imbalance and Lack of explainability	PlantVil- lage-TomEBD and PlantVillage-BBLS	97.2%
[20] , 2023	DenseNet-201	Unclear dataset partitioning, Reliance on DenseNet-201 increases resource demands, and Overfitting risks.	PlantVillage dataset	97.2%
[21], 2024	CNN, ResNet101, Inception v2, and DenseNet121	The study lacks critical implementation details and provides an insufficiently detailed methodology. Furthermore, it fails to present novel contributions, primarily replicating existing frameworks without substantive innovation.	Cotton disease dataset	99.00%
[22], 2024	CFSA-TL-based CNN with LeNet	The model's shallow LeNet architecture lacks advanced features like batch normalization or dropout, limiting its generalization ability and increasing the risk of overfitting and vanishing gradients.	Colored PlantVillage dataset	95.7%
[23], 2025	EfficientNet-B0,EfficientNet-B3, ResNet50, and DenseNet201	The study used standalone CNN models without combining their outputs, showed weak performance, lacked innovation in feature extraction or classification, and relied on unverified, web-scraped images collected under inconsistent conditions.	PlantDoc dataset and Web-sourced dataset	EfficientNet-B3 (80.19%)
[24], 2025	Vgg16, Inception-V3, DenseNet201, and ViT.	The approach incurs high training costs and faces optimization challenges due to gradient instability in ViT. Furthermore, ViTs inherently require large-scale datasets for effective training, yet the available dataset was limited, compounding the issue as no data augmentation techniques were applied.	PlantVillage dataset (Corn and Apple leaf disease)	Apple (99.24%) Corn (98%)

3.3. SQUEEZE AND EXCITATION (SE) NETWORK

The SE network is an attention mechanism used to improve the representational power by modeling the interdependencies between the channels of its convolutional features. The SE network begins with a squeeze operation, where global average pooling is applied to the feature maps output by the preceding convolutional layers. This operation condenses the spatial dimensions (height and width) of each feature map into a single value, effectively summarizing the global information of each channel [31] . Following the squeeze operation, the SE network implements the excitation operation. This involves two fully connected (dense) layers with an ReLU activation function in between. The first dense layer reduces the channel dimension to a bottleneck, capturing the interdependencies between channels. The second dense layer restores the original channel dimension, outputting a set of weights for each channel. The weights obtained from the excitation phase are used to recalibrate the original feature maps. Each feature map channel is scaled by its corresponding weight, allowing the network to emphasize or suppress specific features dynamically based on their importance to the current task [32]. Fig. 2 shows the SE block [7].

3.4. Proposed Model

In this section, we introduce our proposed framework, named the CNN-SE model, which consists of three modules. The local attention features are obtained with the help of the CNN module, while the SE module extracts the global relations from the extracted features, potentially enhancing the learning process to a greater extent. The classification module then classifies the fig leaves as infected or healthy. The flow of the proposed CNN-SE framework is illustrated in Fig. 3. The preprocessing stage prepares the dataset for feature extraction and classification. This involves three steps:

- Class Balancing: Ensuring uniform sample sizes across all classes to mitigate bias.
- Image Resizing: Adjusting images to the standard input size required by the CNN variant used in the proposed model.
- Data Augmentation: Expanding the dataset size through transformations (e.g., rotations, flips) to improve classification accuracy and model generalizability.

A. CNN Module

The CNN block employs convolution layers to learn the characteristics of the input image and extract valuable features. These layers apply convolutional filters to the input image to extract features such as edges, textures, and patterns. Each convolutional layer typically follows ReLU activation function and is often followed by a pooling layer to reduce the spatial dimensions and computational load. In this module, we utilized three pretrained CNN architectures, MobileNetV2, InceptionV3, and Xception, separately. The architecture of these models was modified to improve their ability to learn disease-spot features in fig leaf images. The original classification layers at the end of the pre-trained models were removed, and an SE block was added after the CNN block. The features extracted by the CNN module were then passed into the SE module for further enhancement.

B. SE Module

SE module uses the SE network to further enhance the representational capability of our approach since it captures the interdependence between the channels of the convolutional features of the different layers of the network. It applies the squeeze operation to acquire the channel-wise global context and then applies the excitation operation to address the issue of inter-channel dependencies. In particular, the weights coming from the excitation phase are used to update the original feature maps. This process helps the network to focus on the features that are informative and at the same time reduce other features that are not very useful, thus increasing the representational capability of the model. The squeeze operation sums the feature maps along the spatial domain and this is used one generate a channel descriptor. This is usually done through what is called global average pooling.

$$z_c = \frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} x_{i,j,c} \tag{1}$$

Where $x_{i,j,c}$ is the value at the spatial location (i, j) of the c-th channel of the feature map X with spatial dimensions $H \times W$.

The excitation operation captures the channel-wise dependencies using a simple gating mechanism. This involves passing the squeezed features through two fully connected (FC) layers with ReLU and sigmoid activations, respectively.

$$s = \sigma\big(W_2 \delta(W_1 z)\big) \tag{2}$$

Where z is the sqeeze feature factor of size $C \times 1$ (where C is the number of channel), $W1$ and $W2$ are the weight matrices of the fully connected layers, δ denotes the ReLU activation function, and σ denotes the sigmoid activation function.

The recalibration of the original feature map X is performed by channel-wise multiplication of the original features with the activations from the excitation operation.

$$x'_{i,j,c} = s_c \cdot x_{i,j,c} \tag{3}$$

Where S_c is the excitation output for the c-th channel, and $x'_{i,j,c}$ is the recalibrated feature map.

In general, the SE block can be represented by the following sequence of operations:

$$X' = X \cdot \sigma\Big(W_2\big(\delta W_1\big(GAP\ (X)\big)\big)\Big) \tag{4}$$

C. Classification Module

This module consists of set of layers to train the weights obtained from the SE block. These layers are:

- The Flatten layer: transforms feature maps into a 1D vector.
- Fully Connected (Dense) Layers: Following the flattening layer, there are two fully connected layers. Each dense layer is depicted with its size and activation function:
- Dense Layer (1024, ReLU): A dense layer with 1024 neurons and a ReLU (Rectified Linear Unit) activation function.

- Dropout Layer (30%): A dropout layer with a 30% dropout rate, used to prevent overfitting by randomly setting a fraction of input units to 0 during training.
- Dense Layer (1024, ReLU): Another dense layer with 1024 neurons and a ReLU activation function.
- SoftMax Layer: Assigns class probabilities for the two types of fig leaves: healthy and inflected.

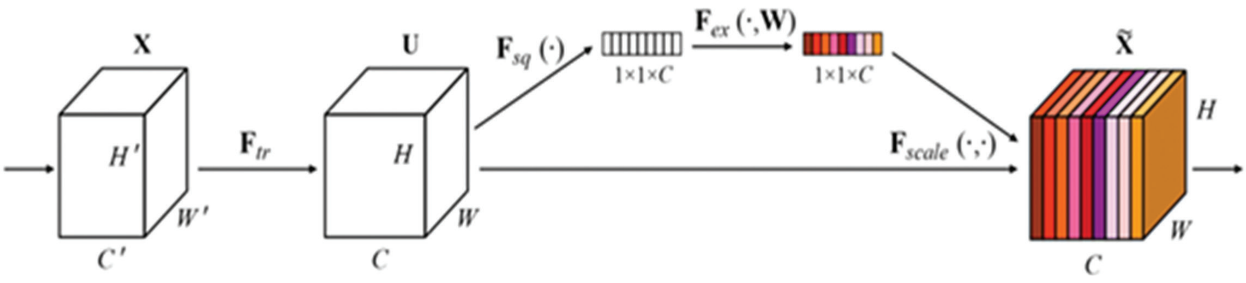

Fig. 2. SE block

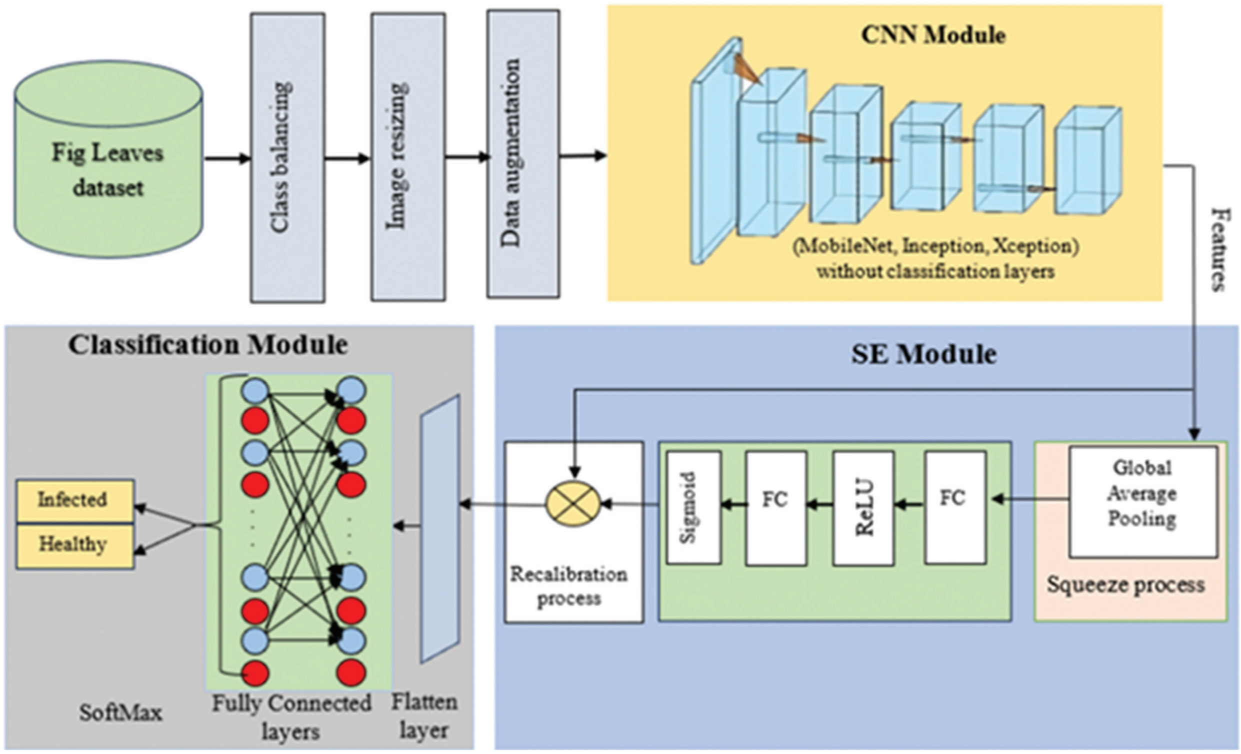

Fig. 3. A schematic diagram of the CNN-SE framework

3.4. Explainable Tools

DL has often been seen as a complex and opaque process, frequently referred to as a "black box" due to the challenges in understanding why a model makes certain decisions. This lack of transparency can undermine trust in the system's final outcomes [12]. To address this issue, this paper utilizes Grad-CAM and t-SNE visualization techniques to overcome these limitations and provide a clearer understanding of how deep learning methods reach their conclusions.

- The Grad-CAM (gradient-weighted class activation mapping) technique is a visualization method that helps in understanding network predictions by creating visual representations of what the network is focusing on. It uses the gradients of the classification score with respect to the final convolutional feature map to identify the parts of an input image that have the most impact on the classification score. Areas with large gradients indicate where the final score relies the most on the data. This technique translates network behavior into interpretable output, which can be used to answer questions about the network's predictions [33]. In this study, Grad-CAM was used to identify the regions of interest emphasized by individual CNN models to better understand the specific traits and features that these models prioritize during detection.
- t-SNE, or t-Distributed Stochastic Neighbor Embedding, is a non-linear dimensionality reduction technique that maintains the data's structure

across different scales [10]. It excels at visualizing high-dimensional datasets by creating a low-dimensional representation that can be plotted. This allows for the visualization of clusters, patterns, and relationships that are challenging to detect in high-dimensional space.

4. RESULTS AND DISCUSSION

4.1. PERFORMANCE EVALUATION METRICS

Testing the proposed model is essential to evaluating its performance. Accuracy, recall, precision, and F1 score are the evaluation metrics are used to evaluate our models. The choice of evaluation metrics is guided by specific criteria. For balanced datasets, accuracy is the most suitable metric. In contrast, for imbalanced data, precision, recall, and the F1-score are more appropriate. Precision and recall help identify specific errors (e.g., false positives and false negatives, respectively), while the F1-score provides a balanced assessment by harmonizing these two metrics. The accuracy measures the proportion of correctly classified samples out of all samples submitted to the model. The recall reflects the model's ability to identify positive samples, indicating how many actual positives were correctly detected. The Precision, on the other hand, measures the proportion of correctly predicted positive samples out of all predicted positives. The F1 Score evaluates the balance between recall and precision in a classification model. Equations (5), (6), (7), and (8) are used to calculate accuracy, recall, precision, and F1 score, respectively. In these formulas, true positive (TP) and true negative (TN) represent correct predictions, while false positive (FP) and false negative (FN) represent incorrect ones [34].

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{5}$$

$$Recall = \frac{TP}{TP + FN} \tag{6}$$

$$Precision = \frac{TP}{TP + FP} \tag{7}$$

$$F1\ score = 2 \times \frac{Precision\ \times Recall}{Precision + Recall} \tag{8}$$

During the model training process, we used specific hyperparameters that were selected through a process of trial and error to ensure optimal model performance. These included a learning rate (0.001), Adam optimizer, batch size (32), 30 epochs, dropout rate (0.3), and a dense layer with 1024 neurons.

4.2. EXPERIMENTAL RESULTS USING CNN MODELS

The experimental results using CNN models—MobileNetV2, InceptionV3, and Xception—are summarized in Table 1 and illustrated in Fig. 4. Each model demonstrated varying degrees of performance in classifying fig leaves as either healthy or infected.

MobileNetV2 achieved the highest overall performance among the three models, with an accuracy of 90.74%. The high recall rate of 95.18% indicates that the model is very effective at identifying true positive cases of infected leaves. The precision of 87.41% suggests that there are some false positives, but overall, the model balances well between precision and recall, leading to a strong F1 score of 91.13%. InceptionV3 also performs well, with an accuracy of 88.70%. Similar to MobileNetV2, it has a high recall rate (95.18%), indicating its strong ability to detect infected leaves. However, its precision is slightly lower at 84.26%, suggesting more false positives compared to MobileNetV2. The F1 score of 89.39% reflects a good balance between precision and recall, albeit slightly lower than MobileNetV2. Xception shows the lowest performance among the three models, with an accuracy of 85.55%. Despite its lower accuracy, Xception has the highest recall rate (95.92%), indicating that it is very good at identifying infected leaves. However, its precision is the lowest at 79.44%, meaning it has a higher rate of false positives compared to the other models. The F1 score of 86.91% is also the lowest, reflecting the trade-off between its high recall and lower precision. The confusion matrices for each model, as illustrated in Fig. 4, show the distribution of true positive, true negative, false positive, and false negative predictions.

Table 1. Performance Metrics of Original CNN Models on Fig Leaf Dataset

CNN model	Accuracy	Recall	Precision	F1 Score
MobileNet	90.74%	95.18%	87.41%	91.13%
Inception	88.70%	95.18%	84.26%	89.39%
Xception	85.55%	95.92%	79.44%	86.91%

4.3. EXPERIMENTAL RESULTS USING CNN-SE MODEL

The experimental results using the proposed model, which integrate CNN architectures with SE blocks, are summarized in Table 2 and illustrated in Fig. 5. Three CNN models—MobileNet, Inception, and Xception—were used separately in the feature extraction phase of the image data, CNN module. SE blocks are added to enhance the representational power based on the assumption that the correlations between the channels of convolutional features require proper modeling. MobileNet-SE model achieved the highest performance among the three proposed models, with an accuracy of 92.90%. The recall rate is 94.81%, indicating a high ability to correctly identify true positive cases of infected leaves. The precision is 91.42%, suggesting a balanced handling of false positives. The F1 score of 93.09% reflects a strong balance between precision and recall, making this model the most robust of the three. Inception-SE model also performed well, with an accuracy of 91.48%. It has a recall rate of 91.48%, showing that it can effectively identify infected leaves. The precision is very close, at 91.50%, indicating a minimal rate of false positives. The F1 score of 91.48% demonstrates a consistent balance between precision and recall, underscoring the model's reliability. While Xception-SE model has the lowest performance among the three proposed models, it still shows substantial improvement compared to the base models. It achieved an accuracy of 89.62%, with a recall rate of 89.62%, indicating good detection of infected leaves. The precision is 89.70%, suggesting effective handling of false positives.

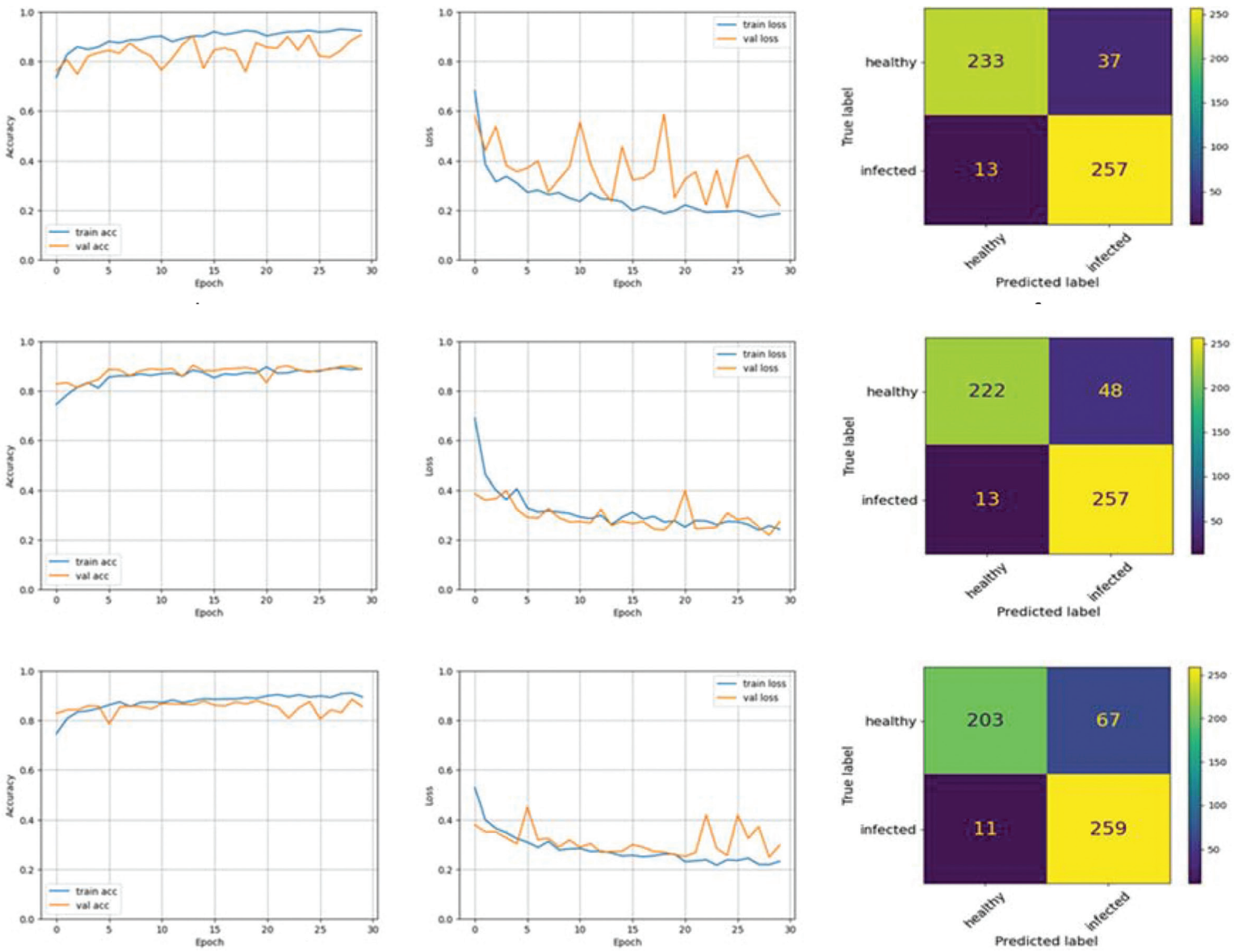

Fig. 5. (**a**) Training Accuracy for MobileNet-SE; (**b**) Loss Curve for MobileNet-SE; (c) Confusion Matrix for MobileNet-SE; (**d**) Training Accuracy for Inception-SE; (**e**) Loss Curve for Inception-SE; (**f**) Confusion Matrix for Inception-SE; (**g**) Training Accuracy for Xception-SE; (**h**) Loss Curve for Xception-SE; (**i**) Confusion Matrix for Xception-SE

The F1 score of 89.62% reflects a well-maintained balance between precision and recall. These results illustrate the effectiveness of the SE blocks in enhancing the performance of CNN models across various metrics, contributing to a more accurate and reliable classification of plants leaves.

Table 2. Performance Metrics of the Proposed CNN-SE Models

CNN model	Accuracy	Recall	Precision	F1 Score
MobileNet	92.90%	94.81%	91.42%	93.09%
Inception	91.48%	91.48%	91.50%	91.48%
Xception	89.62%	89.62%	89.70%	89.62%

4.4. DISCUSSION

The dataset used in this study is the Fig leaf disease dataset. Since this dataset is anonymized and contains no sensitive information (e.g., personal identities and proprietary farm details), there are no ethical concerns regarding data privacy. Furthermore, the training process is conducted offline, eliminating risks associated with unauthorized data sharing or privacy breaches. Regarding environmental impact, our method employs many CNN architectures one of them is a lightweight method and the two others are deep CNN architectures, these methods are optimized for efficiency, which significantly reduces computational demands and energy consumption compared to resource-intensive architectures. This design choice aligns with sustainable practices in AI development. The CNN architectures were selected due to their distinct advantages in achieving the task's objectives:

- MobileNetV2: A highly efficient and lightweight network, offering faster inference speeds compared to bulkier CNNs like VGG and AlexNet.

- InceptionV3: Excels at multi-scale feature extraction by employing parallel convolutional kernels (1×1, 3×3, 5×5) within the same layer, enabling detection of diverse patterns while maintaining lower computational complexity than architectures such as VGG.
- Xception: Optimizes efficiency further by replacing standard convolutions with depthwise separable convolutions—a refinement of Inception's principles—to minimize parameter count and computational overhead.

Integrating these networks with SE block enhances channel-wise feature recalibration, strengthening the model's ability to generalize and improve classification accuracy. Comparing the performance of the original CNN models (MobileNet, Inception, Xception) with their enhanced versions that incorporate SE blocks (MobileNet-SE, Inception-SE, Xception-SE) reveals significant improvements across various metrics. The addition of SE blocks led to better accuracy, precision, recall, and F1 scores for all models. MobileNet-SE showed a notable increase in accuracy from 90.74% to 92.90%, with precision improving from 87.41% to 91.42% and a higher F1 score of 93.09% compared to 91.13%. Inception-SE also benefited from SE blocks, with accuracy rising from 88.70% to 91.48% and precision improving from 84.26% to 91.50%, resulting in a more balanced F1 score of 91.48%. Similarly, Xception-SE exhibited an improvement in accuracy from 85.55% to 89.62%, with precision increasing from 79.44% to 89.70% and a more balanced F1 score of 89.62%. These enhancements highlight the effectiveness of SE blocks in boosting the representational power and overall performance of CNN models for classifying plants leaves. Fig.6 presents the feature distribution visualized using t-SNE for the fig leaves dataset, comparing the original CNN models (left column) and the CNN models enhanced with SE blocks (right column).

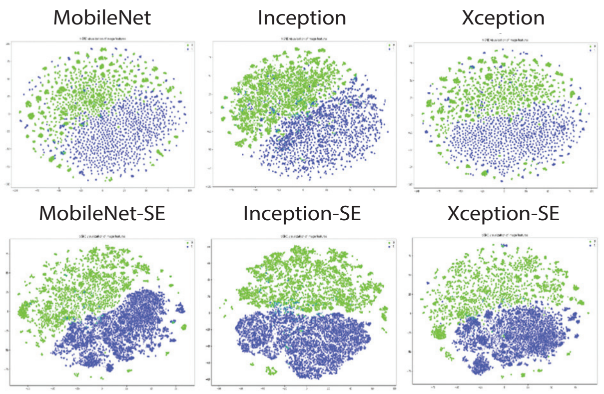

Fig. 6. Feature distribution visualized using t-SNE for fig leaves dataset

The t-SNE plots indicate that the SE blocks have improved the feature separation between healthy and infected leaves, showing more distinct clusters with reduced overlap between the two classes. This suggests better feature representation and classification capability. Fig. 7 illustrates how Grad-CAM can be used to interpret and visualize which parts of an image contribute most to the decisions made by the DL models. The Grad-CAM highlights regions of the leaf that the model considers important for its classification. Brighter areas (in warm colors like red and yellow) indicate regions that have a higher impact on the model's decision, suggesting the presence of disease or other relevant features. This study represents the first application on the Fig leaves dataset, making it challenging to directly compare the performance of the proposed model with existing research in the literature.

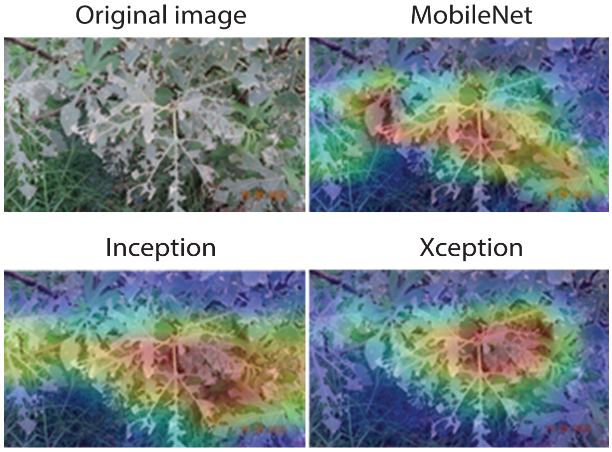

Fig. 7. Grad-Cam with heatmap of infected leaves using CNN models

We did not limit ourselves to these points; instead, we tested our model against multiple other CNN variants and obtained the results shown in Table 3 below.

Table 3. Performance Comparison of the Proposed Model with Other CNN Variants

CNN model	Accuracy	Recall	Precision	F1 Score
EfficientNet	90.3%	86%	96.2	90.9%
InceptionResNetV2	87.96%	89.5%	85.9%	87.7%
ResNet	76.48%	75.6%	78.1%	76.62%

As shown in Table 3, our proposed model demonstrates superior performance compared to the CNN variants listed in the same table, achieving better results across all evaluated metrics. The proposed model's computational efficiency—enabled by its architectures (e.g., MobileNet, Inception, and Xception)—allows it to deliver accurate results more rapidly than traditional CNN-based approaches. This efficiency facilitates seamless integration with portable IoT hardware devices, making it a strong candidate for real-time plant leaf disease detection systems. Such integration represents a promising direction for future research. To demonstrate the model's ability to generalize to unseen data, we employed data augmentation techniques to enhance dataset diversity and mitigate overfitting. Evaluation was conducted on a held-out test set (20% of the data), which was not used during training, and the model consistently achieved high performance across accuracy, precision, recall, and F1-score metrics. Additionally, Grad-CAM visualization confirmed that the model effectively focused on relevant disease regions, further supporting its robustness and interpretability. Despite the significant advantages of our proposed system, certain limitations persist. While SE network enhances feature representation through channel-wise attention mechanisms, the additional parameters it introduces elevate the risk of overfitting when training on small datasets. Moreover, integrating SE network with lightweight architectures like MobileNet or Inception—though beneficial—results in increased computational overhead and inference latency, which may offset the efficiency gains of these architectures. For future directions, applying Vision Transformers (ViT) could enhance the proposed model's accuracy in capturing fine-grained disease patterns, while Generative Adversarial Networks (GANs) could be leveraged to synthetically expand the dataset, addressing limitations in data diversity or scarcity.

5. CONCLUSION

The study presented a novel approach for detecting plant diseases in fig leaves by integrating Squeeze-and-Excitation (SE) networks with pre-trained Convolutional Neural Network (CNN) models, namely MobileNetV2, InceptionV3, and Xception. The proposed CNN-SE framework demonstrated significant improvements in classification accuracy, achieving 92.90%, 91.48%, and 89.62% for MobileNet-SE, Inception-SE, and Xception-SE, respectively. These results highlight the effectiveness of SE blocks in enhancing feature representation and model performance by dynamically recalibrating channel-wise feature weights. Key contributions of this research include addressing data scarcity through transfer learning and data augmentation, improving model interpretability using Grad-CAM and t-SNE visualization tools, and providing a robust solution for sustainable agriculture. The framework's lightweight design ensures computational efficiency, making it suitable for deployment in resource-constrained environments. Despite its successes, the study acknowledges limitations such as the risk of overfitting with small datasets and increased computational overhead from SE integration. Future work could explore advanced architectures like Vision Transformers (ViT) and Generative Adversarial Networks (GANs) to further enhance accuracy and dataset diversity.

6. REFERENCES

[1] X. Wang, J. Liu, "An efficient deep learning model for tomato disease detection", Plant Methods, Vol. 20, No. 1, 2024, p. 61.

[2] I. Pacal, I. Kunduracioglu, M. Hakki, A. Muhammet, "A systematic review of deep learning techniques for plant diseases", Artificial Intelligence Review, Vol. 57, No. 11, 2024, p. 304.

[3] I. Kunduracioglu, I. Pacal, "Advancements in deep learning for accurate classification of grape leaves and diagnosis of grape diseases", Journal of Plant Diseases and Protection, Vol. 131, No. 3, 2024, pp. 1061-1080.

[4] S. A. Jebur, L. Alzubaidi, A. Saihood, K. A. Hussein, H. K. Hoomod, Y. Gu, "A Scalable and Generalised Deep Learning Framework for Anomaly Detection in Surveillance Videos", International Journal of Intelligent Systems, Vol. 2025, No. 1, 2025, p. 1947582.

[5] A. Saihood, T. Saihood, S. A. Jebur, C. Ehlig-Economides, L. Alzubaidi, Y. Gu, "Artificial intelligence based-improving reservoir management: An Attention-Guided Fusion Model for predicting injector-producer connectivity", Engineering Applications of Artificial Intelligence, Vol. 146, 2025, p. 110205.

[6] L. Alkhazraji et al. "Employing the Concept of Stacking Ensemble Learning to Generate Deep Dream Images Using Multiple CNN Variants", Intelligent Systems with Applications, Vol. 25, 2025, p. 200488.

[7] J. Hu, L. Shen, S. Albanie, G. Sun, E. Wu, "Squeeze-and-Excitation Networks", Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18-23 June 2018, pp. 7132-7141.

[8] L. Wang, J. Peng, W. Sun, "Spatial-spectral squeeze-and-excitation residual network for hyperspectral image classification", Remote Sensing, Vol. 11, No. 7, 2019, p. 884.

[9] G. Pacal, I. Işık, "Utilizing convolutional neural networks and vision transformers for precise corn leaf disease identification", Neural Computing and Applications, Vol. 37, No. 4, 2025, pp. 2479-2496.

[10] L. Alzubaidi et al. "ATD Learning: A secure, smart, and decentralised learning method for big data environments", Information Fusion, Vol. 118, 2025, p. 102953.

[11] L. R. Ali, S. A. Jebur, M. M. Jahefer, B. N. Shaker, "Employing Transfer Learning for Diagnosing CO-

VID-19 Disease.", International Journal of Online and Biomedical Engineering, Vol. 18, No. 15, 2022.

[12] A. S. Albahri et al. "A systematic review of trustworthy and explainable artificial intelligence in healthcare: assessment of quality, bias risk, and data fusion", Information Fusion, Vol. 96, 2023, pp. 156-191.

[13] S. Datta, N. Gupta, "A Novel Approach for the Detection of Tea Leaf Disease Using Deep Neural Network", Procedia Computer Science, Vol. 218, 2022, pp. 2273-2286.

[14] F. Khan, N. Zafar, M. N. Tahir, M. Aqib, H. Waheed, Z. Haroon, "A mobile-based system for maize plant leaf disease detection and classification using deep learning", Frontiers in Plant Science, Vol. 14. 2023, p. 1079366.

[15] M. K. A. Mazumder, M. F. Mridha, S. Alfarhood, M. Safran, M. Abdullah-Al-Jubair, D. Che, "A robust and light-weight transfer learning-based architecture for accurate detection of leaf diseases across multiple plants using less amount of images", Frontiers in Plant Science, Vol. 14, 2023, p. 1321877.

[16] Q. Wu et al. "A classification method for soybean leaf diseases based on an improved ConvNeXt model", Scientific Reports, Vol. 13, No. 1, 2023.

[17] Y. A. Bezabih, A. O. Salau, B. M. Abuhayi, A. A. Mussa, A. M. Ayalew, "CPD-CCNN: classification of pepper disease using a concatenation of convolutional neural network models", Scientific Reports, Vol. 13, No. 1, 2023, p. 15581.

[18] R. Bora, D. Parasar, S. Charhate, "A detection of tomato plant diseases using deep learning MNDLNN classifier", Signal, Image and Video Processing, Vol. 17, No. 7, 2023, pp. 3255-3263.

[19] A. Chug, A. Bhatia, A. P. Singh, D. Singh, "A novel framework for image-based plant disease detection using hybrid deep learning approach", Soft Computing, Vol. 27, No. 18, 2023, pp. 13613-13638.

[20] R. Mahum et al. "A novel framework for potato leaf disease detection using an efficient deep learning model", Ecological Risk Assessment: An International Journal, Vol. 29, No. 2, 2023, pp. 303-326.

[21] A. Nagarjun, "An Advanced Deep Learning Approach for Precision Diagnosis of Cotton Leaf Diseases : A Multifaceted Agricultural Technology Solution", Eng. Technol. Appl. Sci. Res., Vol. 14, No. 4, 2024, pp. 15813-15820.

[22] M. Chilakalapudi, S. Jayachandran, "Multi-classi fi cation of disease induced in plant leaf using chronological Flamingo search optimization with transfer learning", PeerJ Computer Science, Vol. 10, 2024, p. e1972.

[23] M. S. Krishna, P. Machado, R. I. Otuka, S. W. Yahaya, F. Neves, I. K. Ihianle, "Plant Leaf Disease Detection Using Deep Learning : A Multi-Dataset Approach", Multidisciplinary Science Journal, Vol. 8, No. 1, 2025, p. 4.

[24] S. Aboelenin, F. Ahmed, E. Mohamed, M. Eltoukhy, "A hybrid Framework for plant leaf disease detection and classification using convolutional neural networks and vision transformer", Complex & Intelligent Systems, Vol. 11, No. 2, 2025, p. 142.

[25] S. J. Hafi et al. "Image dataset of healthy and infected fig leaves with Ficus leaf worm", Data in Brief, Vol. 53, 2024, p. 109958.

[26] D. M. Asriny, S. Rani, A. F. Hidayatullah, "Orange Fruit Images Classification using Convolutional Neural Networks", IOP Conference Series: Materials Science and Engineering, Vol. 803, No. 1, 2020, p. 012020.

[27] S. A. Jebur, M. A. Mohammed, D. H. Abd, L. R. Ali, "MIX - Hybrid Convolutional Neural Network Framework with Explainable Artificial Intelligence for Fig Leaves Disease Detection", International Journal of Intelligent Engineering & Systems, Vol. 18, No. 4, 2025, pp. 881-895.

[28] F. M. J. M. Shamrat, S. Azam, A. Karim, K. Ahmed, F. M. Bui, F. De Boer, "High-precision multiclass classification of lung disease through customized MobileNetV2 from chest X-ray images", Comput. Biol. Med., Vol. 155, 2023, p. 106646.

[29] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, Z. Wojna, "Rethinking the inception architecture for computer vision", Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27-30 June 2016, pp. 2818-2826.

[30] L. Alzubaidi et al. "MEFF - A model ensemble feature fusion approach for tackling adversarial attacks in medical imaging", Intelligent Systems with Applications, Vol. 22, 2024, p. 200355.

[31] I. Pacal, "Enhancing crop productivity and sustainability through disease identification in maize leaves : Exploiting a large dataset with an advanced vision transformer model", Expert Systems with Applications, Vol. 238, 2024, p. 122099.

[32] J. Chen, D. Zhang, M. Suzauddola, Y. A. Nanehkaran, Y. Sun, "Identification of plant disease images via a squeeze-and-excitation MobileNet model and twice transfer learning", IET Image Processing, Vol. 15, No. 5, 2021, pp. 1115-1127.

[33] R. R. Selvaraju, A. Das, R. Vedantam, M. Cogswell, D. Parikh, D. Batra, "Grad-CAM: Why did you say that?", arXiv:1611.07450, 2016.

[34] S. A. Jebur, K. A. Hussein, H. K. Hoomod, L. Alzubaidi, "Review on Deep Learning Approaches for Anomaly Event Detection in Video Surveillance", Electronics, Vol. 12, No. 1, 2023, p. 29.