GMR

The Original

A HYBRID OPTIMIZATION AND DEEP LEARNING APPROACH FOR CROP DISEASE DETECTION AND PESTICIDE RECOMMENDATION

[1*]Preeti Shukla, [2]Dr. Amit Kumar Chandanan

[1]Research Scholer, Department of Computer science and Information Technology Guru Ghasidas Vishwavidyalaya, Bilaspur (CG), India, https://orcid.org/0009-0004-1134-8085, Mail id: preetishukla5454895@gmail.com
[2]Associate Professor, Department of Computer science and Engineering Guru Ghasidas Vishwavidyalaya, Bilaspur (CG), India. https://orcid.org/0000-0002-3269-5383

ABSTRACT
Crop diseases remain a major threat to global agriculture, food security, and economic stability, making rapid and accurate detection essential for effective management. Traditional diagnostic methods are often time-consuming, labor-intensive, and inadequate for addressing newly emerging complex diseases. To address these challenges, this study proposes a Crop Ensemble Net approach for disease detection and pesticide recommendation. The method to apply advanced preprocessing techniques to improve the resolution of images and identify distinguishing features between diseased and healthy leaves. Feature selection and optimization are further improved using Mud Ring Optimization (MRO) and Secretary Bird Optimization (SBO), which strengthen detection accuracy. The proposed ensemble integrates three high-performing deep learning architectures—Efficient Net, ResNet, and InceptionV3—to balance speed, precision, and multi-scale feature extraction. Efficient Net ensures computational efficiency, ResNet captures intricate hierarchical structures, and InceptionV3 identifies disease patterns across varying scales. Experimental results demonstrate that the Crop Ensemble Net achieves superior accuracy and robustness compared to existing methods, offering a scalable and interpretable solution for sustainable crop disease management.

KEYWORDS: Crop Disease; Detection; Optimization; Pesticide Recommendation; Deep Learning.

1. INTRODUCTION

In modern agriculture, effective control of crop diseases is essential for improving yield and quality while reducing the severe losses caused by plant pathogens. Excessive losses to food crops can occur because of plant diseases, and result in ripple effects contributing to global food insecurity [1, 2]. Although pesticides and insecticides may provide temporary relief, they are not sustainable long-term solutions, as chemical treatments can damage crops and pose risks to human health [3]. Early diagnosis forms the foundation for effective disease control and plays a critical role in safeguarding agricultural production [4]. Early detection of crop diseases, recommending suitable pesticides, and reducing reliance on manual inspection, advanced systems can help farmers minimize losses and enhance productivity [5].

Excessive use of chemical pesticides and fertilizers increases costs, degrades soil, and harms human health. While several pesticides such as copper fungicides, Mancozeb, and Chlorothalonil are commonly used, sustainable practices like pruning, sanitation, biological control, and resistant crop varieties offer safer alternatives [6, 7]. However, the effective transition to these sustainable methods is often hindered by the lack of precise diagnostic tools. Without accurate identification of the specific pathogen, farmers frequently resort to broad-spectrum chemical applications as a precautionary measure, which exacerbates environmental degradation [8] [9]. Traditional methods of crop disease detection primarily rely on manual inspection by farmers or agricultural experts, which is inherently time-consuming, labor-intensive, and often subjective [10]. These methods become particularly inefficient in large-scale farming operations, where timely intervention is crucial to prevent widespread crop damage [11].

Recent studies have highlighted significant advancements in plant disease detection and classification through image processing, machine learning, and deep learning techniques. In particular, numerous computer vision (CV) approaches have been developed to enable the automatic recognition and detection of diseased plants [12] [6]. However, many of these methods depend heavily on handcrafted feature identification, which requires substantial expertise in image processing and limits scalability [13]. Furthermore, many current systems are heavily focused on the identification of plant diseases and do not provide actionable recommendations to the farmers themselves. They often lack adaptability across different crop species, struggle with real-time deployment in field conditions, and fail to integrate multimodal data such as soil health or weather patterns [14] [15]. To address these shortcomings, the proposed ensemble model consisting of EfficientNet, ResNet, and InceptionV3 generates highly accurate predictions of plant diseases, while the integration of Graph Convolutional Networks (GCN) enables context-aware pesticide recommendations. This framework not only improves prediction accuracy but also reduces reliance on manual expertise, supports sustainable crop management, and offers practical guidance that can be extended to diverse agricultural environments. The key contributions of this study are summarized as follows:

- A novel combination of Mud Ring Optimization (MRO) and Secretary Bird Optimization (SBO) is introduced to enhance crop disease detection by improving classification performance and feature selection efficiency.
- A weighted probability fusion approach is employed to integrate EfficientNet, ResNet, and InceptionV3, thereby improving accuracy, robustness, and generalization across diverse agricultural diseases.
- Beyond disease detection, a Graph Convolutional Network (GCN) is incorporated to map illness predictions to crop–disease–pesticide interactions, enabling context-aware and practical treatment suggestions for farmers.

The remainder of this paper is organized as follows: Section 2 presents related research and literature review, Section 3 introduces the proposed framework, Section 4 discusses experimental results and analysis, and Section 5 concludes with final observations and implications of this study.

2. LITERATURE REVIEW

Vijayan and Chowdhary [16] introduced a hybrid bio-inspired algorithm called Hybrid WOA_APSO, which combines Adaptive Particle Swarm Optimization (APSO) and the Whale Optimization Algorithm (WOA) for rice crop disease. This approach utilized Convolutional Neural Networks (CNNs) for classification. Their study emphasized the importance of preprocessing, segmentation, and feature extraction, noting that effective feature selection is essential for improving classification performance.

Li et al. [17] developed an integrated pest and disease recognition system using agricultural drones and deep learning. The system applies convolutional neural networks (CNNs) along with high-quality image capture and wirless data transfer to identify and classify different pests and diseases. Their framework includes modules for image acquisition, data transmission, and recognition, allowing fast and reliable detection even in remote areas.

Bilal et al. [18] suggested a deep fusion of MobileNetV2 and EfficientNetB0 using Deep Learning to identify pests and diseases in agriculture in real-time. Their approach emphasized deployment in edge environments through techniques like pruning and quantization, enabling use on low-power devices such as smartphones, Raspberry Pi, and drones. The system concluded that lightweight deep learning models can provide scalable and cost-effective solutions for early detection, supporting sustainable agriculture and food security.

Shafik et al. [19] emphasizes the importance of Artificial Intelligence and Deep learning in improving precision agriculture, focusing on early detection of plant diseases to protect crop yields, utilizing the Turkey Plant Pests and Diseases dataset and the applied of a ResNet-9 Model for Classification. A key contribution was the use of explainable AI techniques, such as SHAP saliency maps, which provided visual insights into how the model made predictions.

Banothu et al. [20] introduced a deep learning model, named DenseNet for automated disease recognition and pesticide recommendation. Mainly utilized for early detection of plant diseases caused by bacteria and fungi, which reduce crop yield and quality. In many regions, disease identification is still done visually, making it slow and costly. Their approach was validated by experts and shown to be effective.

Wang et al. [21] proposed an innovative crop disease identification model, named DC-GAN-MDFC–ResNet, which combines a deep convolutional generative adversarial network with a multidimensional feature compensation residual network. This approach was designed for fine-grained identification of common rice disease including bacterial leaf blight, leaf streak, and panicle blight. Their approach included preprocessing steps like data improvement, normalization, and singular value decomposition to enhance training quality.

Bouacida et al. [22] introduced a CNN-based Inception model that uses deep learning techniques to classify healthy versus diseased leaves of several different types of crops. The two point emphasis of their approach was on locating small areas of diseased tissue on the leaf surface and determining the number of diseased leaves within each area. By employing a compact CNN-Inception architecture, the system efficiently processes small regions while maintaining strong performance.

Ritharson et al. [23] examined CNN and transfer learning techniques to classify rice leaf diseases. The dataset consists of rice leaf sample images and is classified into nine categories — healthy (no disease) and several diseased states – such as blight, tungro, blast and brown spot. The dataset was created, labeled and validated by experts to be accurate and complete; augmentation was used to increase the number of images in the dataset. The study tested custom CNN architectures along with pre-trained networks such as VGG16, Xception, ResNet50, DenseNet121 and various models using Inception as a base for classification.

Li et al. [24] proposed a dual-branch deep learning architecture that captures crop disease features from b the frequency and spatial domains. The frequency branch of the method extracts disease-related features from frequency domain data (spectrum) while the deformable attention transformer branch captures both global and local features from the spatial domain. After that, introduced a Multi-Spectral Channel Attention Fusion (MSAF) method as well as an improved bias loss function to optimally combine these two branches. The model is lightweight, with relatively few parameters, making it suitable for complex agricultural environments. Table 1 presents the comparative analysis of Deep Learning Approaches in Crop Disease Detection.

Table 1: Comparative summary of recent deep learning approaches for crop disease

Study	Aim	Method	Advantages	Limitations
Li et al., 2024	Develop integrated pest/disease recognition for drones	CNN + HD image acquisition + wireless transmission	Real-time detection, improved accuracy, remote operation	Limited to drone-based imaging; generalization across crops not tested
Bilal et al., 2025	Real-time pest/disease detection in precision farming	Fusion of MobileNetV2 + EfficientNetB0, edge deployment	High accuracy (89.5%), optimized for smartphones/Raspberry Pi, cost-effective	Dataset limited to 4 crops; early-stage detection not emphasized
Shafik et al., 2025	Early detection of plant diseases with interpretability	ResNet-9 + SHAP saliency maps	Very high accuracy (97.4%), interpretable predictions	Dataset imbalance; limited crop diversity
Banothu et al., 2024	Early detection of bacterial/fungal diseases	DenseNet CNN model + cloud service	Accuracy >92%, scalable, accessible for farmers	Training requires large parameters; cloud dependency
Wang et al., 2024	Fine-grained rice disease detection	DC-GAN + MDFC-ResNet	High accuracy (95.9%), stable training, segmentation-free	Focused only on rice diseases; preprocessing complexity
Bouacida et al., 2025	Generalizable detection across crops	Small CNN-Inception model	Accuracy up to 97.1%, strong generalization	Focus on leaf regions only; prevalence estimation may miss whole-plant context
Ritharson et al., 2024	Classify rice leaf diseases	Custom CNN + VGG16, ResNet, DenseNet comparisons	Exceptional accuracy (99.9%), interpretable via feature visualization	Limited to rice dataset; manual labeling effort
Li et al., 2024	Integrate frequency + spatial features	Dual-branch CNN + Transformer + MSAF fusion	High accuracy (96.7%), lightweight (14M params), robust	Tested only on benchmark dataset; real-world validation missing

2.1. Research Gap

Despite significant progress in crop disease detection through deep learning, hybrid optimization, drone-based monitoring, lightweight CNN models, and explainable AI, most existing studies remain focused on improving classification accuracy rather than providing actionable treatment recommendations. Current approaches often rely on specific datasets and struggle to generalize across diverse crops and environments, while large-scale validation in real-world farming conditions is limited. Moreover, many models function as "black boxes," offering little interpretability for farmers and agronomists, and early-stage detection before visible symptoms remain a challenge. To overcome these gaps, our proposed technique integrates a dual-branch deep learning framework that combines spatial and frequency domain features with advanced fusion and optimized loss functions. This approach not only enhances accuracy and robustness in complex environments but also ensures lightweight deployment, interpretability, and practical usability, thereby offering a scalable solution for sustainable and precision agriculture.

3. PROPOSED METHODOLOGY

This study presents the first application of a hybrid feature selection technique that combines Secretary Bird Optimization (SBO) and Mud Ring Optimization (MRO) in crop disease identification. Weighted probability integration is used to integrate an ensemble of EfficientNet, ResNet, and InceptionV3 to enhance generalization. By integrating a Graph Convolutional Network (GCN) for pesticide recommendation, we bridge the gap between diagnostic and practical therapy, in contrast to previous research that terminates at classification. Reliability in real-world scenarios is further guaranteed by a strong preprocessing pipeline. Figure 1 shows the overall architecture of the proposed methodology.

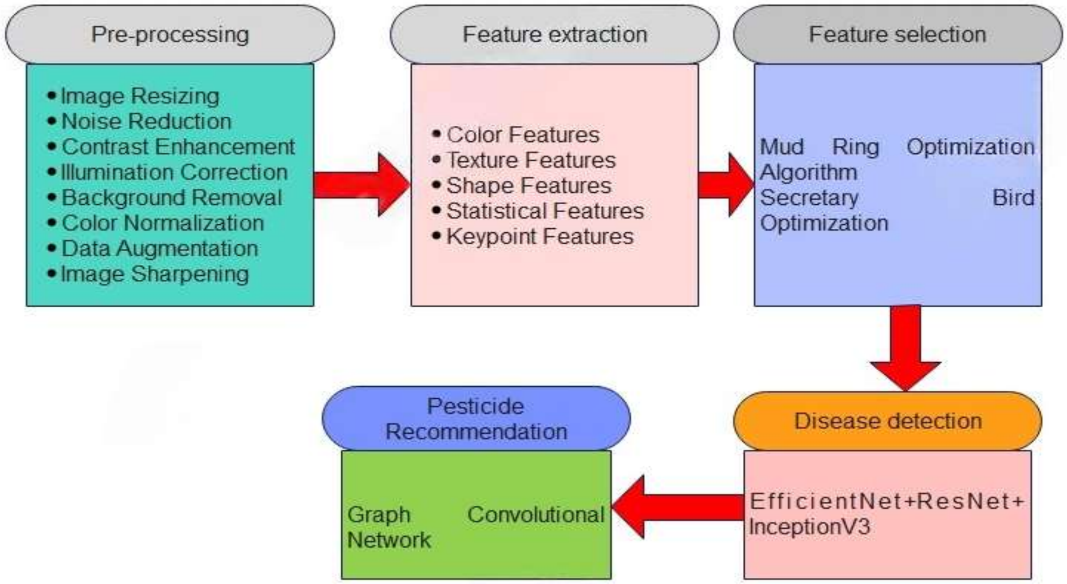


Figure 1: Architecture of the Proposed Methodology

3.1. Pre-processing

3.1.1. Image Resizing

In order to have the same size of images, it is important for datasets to apply a standard size. Neural networks require a constant input size. Therefore, the image is resized to 256×256.

3.1.2. Noise Reduction

Noise emanates from various sources, including camera sensors or the environment, and filters can aid in minimizing it. Image smoothing is done using a Gaussian function, while median filtering involves substituting a pixel's value with the median around it so that we eliminate salt and pepper noise.

3.1.3. Contrast Enhancement

In the image, such a transformation makes the difference between dark and light areas more pronounced. Adaptive Histogram Equalization (AHE) is designed to even out contrast, especially in small zones of an image, which allows us to examine disease symptoms more efficiently.

3.1.4. Illumination Correction

A homomorphic filter is used to fix non-uniform lighting by examining the image in the frequency space. This allows one to extract the blocked part from the well-lit areas and vice versa, thus enhancing visibility.

3.1.5. Background Removal

For a better focus on our area of interest, it is important to isolate a leaf from its environment. Such techniques as Canny Edge Detection are used to highlight the edge of a leaf, hence making it possible to accurately extract a leaf from its background.

3.1.6. Color Normalization

Ensuring color distribution is similar in all images, irrespective of variations in light sources, is paramount for unbiased analysis. This includes changing pixel values to fit within a certain range of values (e.g., 0 to 1 or -1 to 1).

3.1.7. Data Augmentation:

In order to improve our model's strength and adaptability, we generate augmented images through the application of transformations such as rotation, flipping, scaling, or cropping, thereby artificially increasing our training dataset's size, which introduces more variability.

3.1.8 Image Sharpening:

Sharpening methods enhance the edges and finer details of the image, which increases the visibility of symptoms of diseases such as spots or lesions. The unsharp masking technique removes the blurred version of the image from the original to emphasize its edges.

3.2 Feature extraction

A crucial step in disease detection systems for crops is the extraction of features. It is the process of selecting significant attributes in pictures of crop leaves that distinguish between healthy leaves and those with illnesses, as well as identifying the cause of illness. Measurement tools for image analysis convert the consciously observed patterns from raw data into quantities that can be used by machine algorithms for more effective disease recognition in plants.

3.2.1. Color Features

Different colors signify alterations under specific circumstances when an area changes its hue and develops blotches. To identify these changes, color properties are used. Statistical colour moments such as average value, variance, and skewness give a condensed view of this pattern. They can also be transformed into different color models like RGB, HSV, HLS, or Lab systems, ensuring uniformity of color tones within different regions forever.

3.2.2. Texture Features

Features like smoothness, roughness, or uniformity recognize the textural patterns on a surface. Texture information is captured by the Gray-Level Co-occurrence Matrix (GLCM), which counts pixel pairs with specific values. Local Binary Patterns (LBP) create a binary pattern for each pixel based on local texture information. Gabor filters are used to analyze multi-resolution textures depending on certain frequency or orientation values.

3.2.3. Shape Features

Shape properties encompass the size and shape of leaves and damaged sections. Identification of the boundary serves as a sketch outline of the affected portions. Area, perimeter, eccentricity, and aspect ratio are distinct state quantities.

3.2.4. Statistical Features

Information about gradient directions is captured by HOG (histogram-oriented gradients), which reveals the edge orientation. Apart from shape-related information, moments like spatial, central, and Hu moments measure the intensity distribution in an image as well.

3.2.5 Key point Features

The SIFT system is in charge of spotting and describing set points in the image that do not change at all when the size, rotation, or brightness circumstances are changed.

3.3 Feature Selection

After feature extraction, the important features are selected by combining the Mud Ring and the secretary bird optimization algorithm.

3.3.1. Mud Ring Optimization Algorithm

The Mud Ring Algorithm is designed similarly to how bottlenose dolphins search for food together in the presence of a school of fish around the area where these animals are known to exist; thereafter, they form mud rings for feeding. After initializing the population, the fitness of each solution is determined using Eqn. (1).

$$Fitness = min(Error) \tag{1}$$

(i) Search for prey

If we mimic some of the echolocation skills that dolphins possess, all sharks employ the echolocation technique to estimate the length of the range while seeking food. This particular motion pattern is applied to search for food, and at the same time, when chasing prey, dolphins move randomly using velocities $\vec{W}$ at points with an amplitude of the sound $\vec{L}$ at position $\vec{E}$.

The volume of dolphin sound changes due to the distance from food (depending on how close they are to the prey; dolphins can regulate the beam form of their sound). The change in volume according to time-step, and the pulse rate s that ranges from 0 to 1 over a wide range, but for $s = 0$ It is low-emission pulses while $s = 1$ High emissions pulses vary regionally. The vector $\vec{L}$ is determined as:

$$\vec{L} = 2\vec{b} \cdot \vec{s} - \vec{b} \tag{2}$$

where $\vec{s}$ Signifies a random vector ranging from 0 to 1.

$$\vec{b} = 2\left(1 - \frac{1}{I_{max}}()\right) \tag{3}$$

In a parameter space with multiple dimensions, we use virtual dolphins to move as if they are searching naturally. To find potential takeaways, they delve into this space from random starting points set by their locations relative to each other. As part of the encouragement for them to disperse while identifying deserving candidates, we $\vec{L}$ That varies randomly within values exceeding one or below negative one. Consequently, the marine creatures take divergent paths and try to catch their food.

Rather than selecting the fittest dolphin, we consider selecting a dolphin randomly based on this selection method, which ensures exploration to enable the MRA algorithm to perform a global search of all possible solutions. Below is a mathematical expression of the MRA algorithm, which involves determining the criteria for updating positions and velocities: The expression for the MRA algorithm based on velocity $\vec{W}^t$ at time step t is given as:

$$\vec{E}^t = \vec{E}^{t-1} + \vec{W}^t \tag{4}$$

where W is set as a random vector. Each dolphin is initially given a random velocity within the range of $[W_{\min}, W_{\max}]$.

(ii) Feeding

When dolphins find the prey, they can encircle it. While we do not know the exact best location of the design in the search space a priori, the MRA method suggests that any prey target that is optimal or close to that is considered (at that time) as the best solution. Thus, by adjusting their position, taking a cue from the top dolphin, other dolphins change their locations. The behavior described above can be represented by the following equations:

$$\vec{B} = \left|\vec{D}\vec{E}^{*t-1} - \vec{E}^{t-1}\right| \tag{5}$$

$$\vec{E}^{t} = \vec{E}^{*t-1} \cdot sin(2\pi k) - \vec{L} \cdot \vec{B} \tag{6}$$

In this, t Is the latest source of time, l, chosen randomly? $\vec{D}$, $\vec{L}$ are co-efficient vectors, $\vec{E}$ denotes the dolphin's positional vector and $\vec{E}^{*}$ Refers to the best attained dolphin position until now of the positional vector.

$$\vec{C} = 2 \cdot \vec{s} \tag{7}$$

3.3.2. Secretary Bird Optimization

The exploitation phase from Secretary Bird Optimization is incorporated into the Mud Ring Optimization algorithm for faster convergence.

3.3.2.1 Escape strategy (Exploitation)

The secretary birds will first scout for a good place to hide as soon as they detect any predator in the vicinity. While there is none, they have to choose between flying off and running hastily for refuge. This strategy is balanced through the introduction of a dynamic perturbation factor represented by $\left(1-\frac{I}{I_{\max}}\right)^2$; that is, by randomizing with possible states to reach the goal, which helps not only explore new solutions but also exploit old ones. Modifying these factors may boost exploration or deepen exploitation during different stages. In conclusion, both tactics of evasion employed by secretary birds can be mathematically represented using Eq., whereas Eq. could be used to show the updated condition or state of things.

$$y_{i,j}^{new,P2} = \begin{cases} B_1: y_{best} + (2 \times RB - 1) \times \left(1 - \frac{I}{I_{max}}()^2{}_{i,j}{}_i\right) \\ B_2: y_{i,j} + S_2 \times \left(y_{rand} - Q \times y_{i,j}\right), else \end{cases} \tag{8}$$

$$Y_i = \begin{cases} Y_i^{new,P2}, if F_i^{new,P2} < F_i \\ Y_i, else \end{cases} \tag{9}$$

Here, with $s = 0.5$ equal to 0.5, S_2 signifies the random creation of an array with dimensions $(1 \times Dim)$ from the normal distribution, y_{rand} denotes the random current iteration's candidate solution, and Q Represents the randomly selected integer 1 or 2, which is determined by Eqn. (10).

$$Q = round(rand(1,1) + 1) \tag{10}$$

3.4. Disease Detection

We thoroughly utilized the potentials of EfficientNet, ResNet, and InceptionV3 by combining them in our ensemble models to enhance both the reliability and precision of identifying plant diseases. In particular, the Efficient Net model serves as a means of extracting general features in a manner that balances accuracy with computational efficiency. Additionally, deep residual networks in ResNet enable it to solve the vanishing gradient problem, hence it can learn a very deep network that captures complex hierarchical features. Using inception modules to extract multi-scale features enables InceptionV3 to recognize different disease symptoms effectively. By using these models, we get a more powerful and accurate disease detection mechanism by combining them as a whole.

3.4.1. Efficient Net

Efficient Net utilizes a method for scaling the model that uniformly adjusts depth, width, and resolution using a compound coefficient. This approach helps Efficient Net achieve better accuracy with fewer parameters and reduced computational costs compared to other deep learning models. The network scales up by systematically adjusting its depth, width, and resolution together using fixed coefficients.

The input image is resized to a specific resolution, for instance, like 224×224. The first layer that processes the input image using a standard convolutional layer is the stem block. EfficientNet utilizes primarily MBConv layers; these have depthwise separable convolutions together with pointwise convolutions and activation functions. Compound scaling adjusts the width, depth, and resolution of a network based on a common scaling formula. Global average pooling reduces the spatial dimension of feature maps, while a fully connected layer outputs classification probabilities for each class.

3.4.2. ResNet

Blocks with residual connections that bypass one or more layers were introduced by ResNet. It was implemented in order to resolve the problem of the gradient vanishing while still allowing for deeper networks. Each block consists of convolutional layers with straight skip connections, which basically just add an input to the output of the block. ResNet deeper model versions incorporate bottleneck layers, which are made of a 1x1 convolution, followed by a 3x3 convolution, and another 1x1 convolution. Skipping makes it possible to train much deeper networks by allowing gradients to flow directly through the network.

An input image is resized to a preset resolution, such as (224x224). Immediately after is a convolutional layer with batch normalization and ReLU activation function. Some residual blocks are made of stacked convolutional layers and skip connections. Besides, this layer downsizes the spatial dimensions of the feature maps. Eventually, a dense layer takes care of classification by producing probabilities for every category.

3.4.3. InceptionV3

Inceptionv3 utilizes different sizes of convolution filters to handle the input, allowing it to recognize features on all ends. Then these filters are merged into a different layer. Each inception model has various parallel convolution layers with different filter sizes combined with some pooling activities. To eradicate the disappearing gradient and enhance studying, Auxiliary classifiers are inserted.

The input image is resized to a predetermined resolution. Then a sequence of convolution layers followed by a max-pooling layer is executed on it. Within these inception modules, various convolutions and pooling operations take place simultaneously. Also, the hidden layers generate more gradients. Finally, a layer called a pooling layer compresses the spatial dimensionality, and a fully connected (dense) layer makes a class prediction (dense). Thus, every class may receive some probability. Figure 2 demonstrates the Ensembled model of ResNet, EfficientNet, and Inception v3.

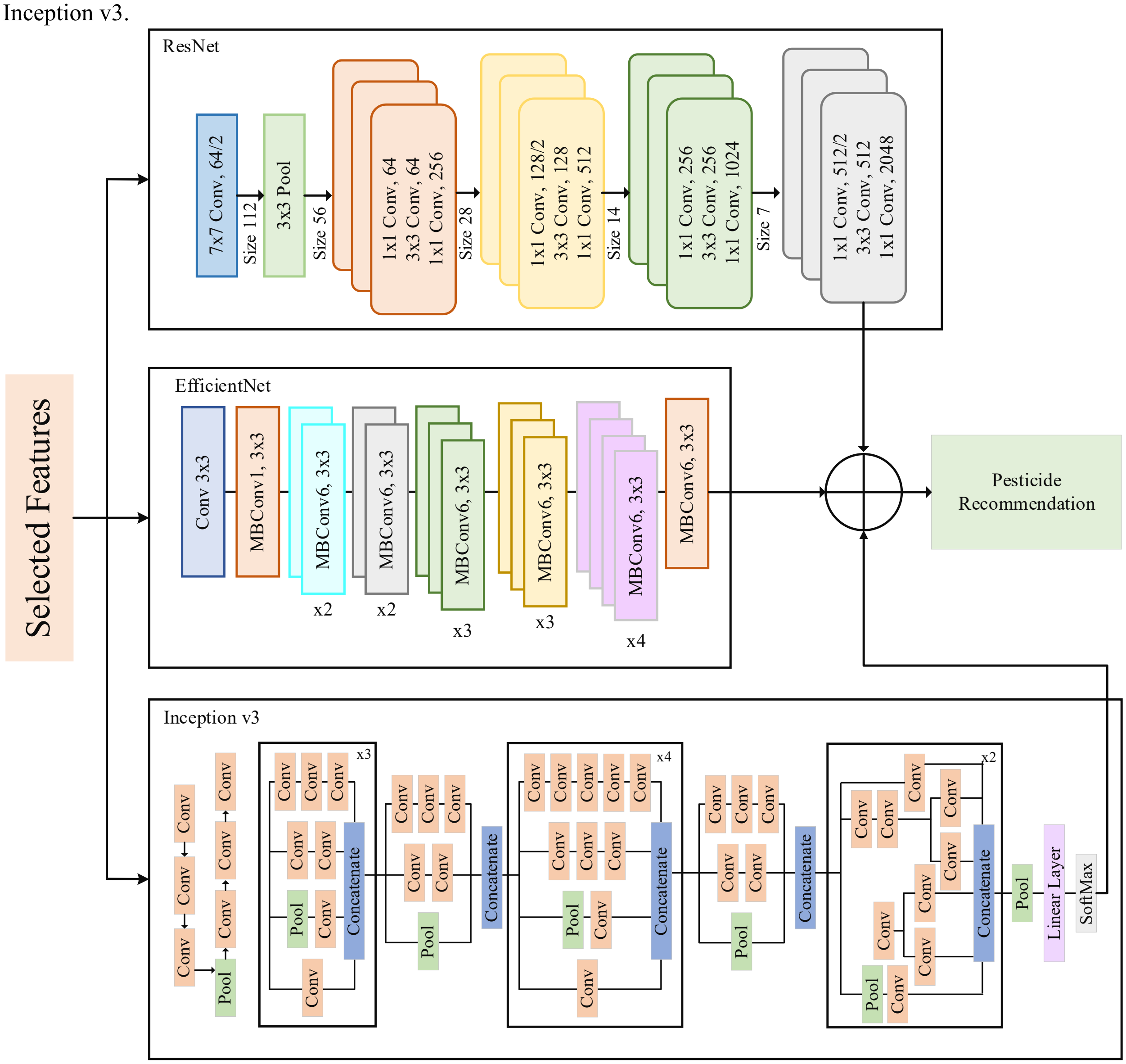


Figure 2: Framework of the Ensembled Model

Feature vector from the EfficientNet $f_E(x)$, ResNet $f_R(x)$, and Inception V3 $f_I(x)$. The combined feature vector is expressed in Eq. (11)

$$F(x) = |f_E(x)||f_R(x)||f_I(x)| \tag{11}$$

Where || Denotes the Concatenation. The vector is passed into a fully connected classifier in following equation (12).

$$\hat{y} = SoftMax(W \cdot F(x) + b) \tag{12}$$

3.5. Pesticide Recommendation

The first thing we need to do is gather data on crops, diseases, pesticides, and environmental conditions that are interconnected. This information is organized into a visual format where crops and diseases each have their node, while disease occurrence in specific crops is linked by edges among other connections (e.g., how effective a pesticide is against a given disease).

The Graph Convolutional Network (GCN) layers pass messages and aggregate features across nodes in the graph. In each GCN layer, data spreads out from neighboring nodes to alter the feature representations of individual nodes based on their local graph structures, enabling the model to comprehend relations and connections of various elements within the agricultural ecosystem.

The GCN model is trained on labeled data in which pesticides were used to control crop diseases. The goal is to use available labeled data and the structure of the graph in order to predict the most effective pesticides that could be used to control new diseases. Such a problem could easily be described as classifying or ranking, where it is necessary for one to determine how much of a given pesticide would help in curing a certain disease. Figure 3 shows the graph convolutional network.

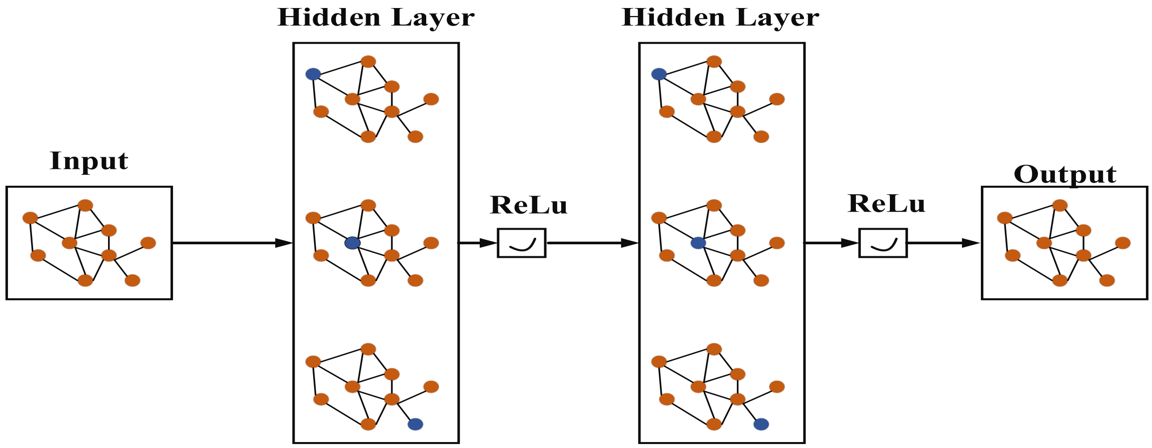


Figure 3: Graph Convolutional Network

Description of the GCN

- **Objective:** Multi-class classification (selecting the right pesticide).
- **Loss Function:** Cross-Entropy Loss.
- **Optimizer:** Adam with learning rate 1e−3.
- **Evaluation Metrics:** Accuracy, F1-score, precision,etc
- **Regularization:** Dropout (0.5) to reduce overfitting.

Algorithm
Input: Crop leaf image I
Preprocess $I \rightarrow I'$ (noise reduction, enhancement, etc.)
Extract handcrafted + deep features. F
Apply MRO + SBO ⟶ select optimal features. $F*$
Pass $F*$ Into EfficientNet, ResNet, InceptionV3 ⟶ get outputs pE, pR, pI
Fuse outputs: $p = \alpha pE + \beta pR + \gamma pI$
Predict disease label $y = \text{argmax}(p)$
Use GCN with (crop, disease, environment) ⟶ recommend pesticide. P
Output: (y, P)

Crop disease detection by manual inspection is slow, arbitrary, and inappropriate for extensive surveillance. Modern computer vision and deep learning techniques, on the other hand, automatically identify minute patterns in photos, increasing precision and repeatability. While lightweight models allow for real-time diagnosis on mobile and edge devices, transfer learning lowers the amount of data needed. Compared to conventional field inspection, these technologies offer faster, scalable, and more dependable detection when combined with IoT and remote sensing.

4. EXPERIMENTAL RESULTS

The proposed method was implemented and simulated in Python, and its performance was evaluated against existing techniques. The dataset was split using stratified sampling under different training scenarios (70/10/20 and 80/10/10), with augmentation applied only to the training set. To ensure a fair comparison, the evaluation considered standard performance metrics such as sensitivity, specificity, precision, recall, F-measure, Matthew's correlation coefficient (MCC), computational complexity, training time, robustness, and scalability, compared with existing sytem.

4.1. Dataset description

With over 54,303 leaf images divided into 38 classes that represent various plant species and their corresponding healthy or diseased states, the PlantVillage dataset serves as a standard collection for plant disease detection. For classification tasks, supervised learning is made possible by the annotation of each image with its matching label. The class distribution in the dataset spans several species, including tomato, potato, maize, and apple, and includes photos of both healthy and damaged leaves. In order to improve model generalization, certain versions further incorporate augmented samples, which increase the dataset to about 61,000 images across 39 categories. These examples are produced using manipulations such as rotation, scaling, and noise addition. This wide and well-balanced dataset is now a common tool for creating and assessing deep learning models for the detection of agricultural diseases [25].

4.2. Experimental description

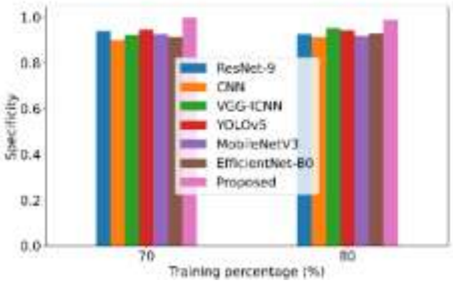
Training, validation, and test sets were randomly selected from the dataset. The data was split into three sets: 70% training, 10% validation, and 20% test sets for the 70% training scenario. In the same way, 80% of the data was utilized for training, 10% for testing, and 10% for validation in the 80% training instance. To maintain class distribution across sets, stratified sampling was used. While the test set was rigorously held out to assess generalization, the validation set was used to fine-tune hyperparameters (learning rate, batch size, optimizer, and ensemble fusion weights). To avoid data leaking into the test and validation sets, data augmentation was solely used on the training set. To ensure consistency in training and evaluation, the dataset was split using stratified sampling, and the optimization algorithm was configured with fixed parameters. Table 2 summarizes the key parameter settings of the proposed system.

Table 2: Hyperparameters of the optimized algorithm

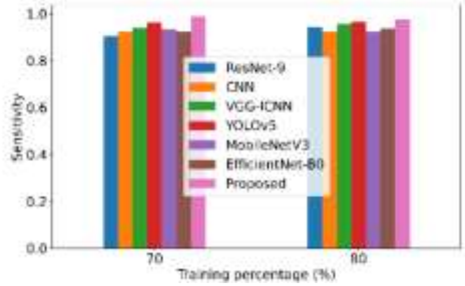
Category	**Hyperparameter**	**Value**
Training	Optimizer	Adam
	Learning Rate	0.001
	Batch Size	32
	Epochs	50
	Validation Split	0.2
	Early Stopping Patience	10
Network	Activation Function	ReLU (Hidden), Softmax (Output)
	Loss Function	Categorical Crossentropy
	Dropout Rate	0.5
	Weight Initialization	He Normal
Regularization	L2 Regularization	0.0001
Optimization Algorithm	Population Size	30
	Number of Iterations	100
	Lower Bound	0
	Upper Bound	1
	Mutation Rate	0.1
	Crossover Rate	0.8

4.3. Performance Evaluations

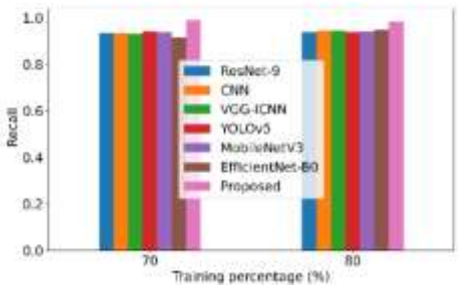
The performance metrics specificity, sensitivity, recall, precision, accuracy, NPV, MCC, FNR, FPR, F-Measure, scalability, computational complexity, robustness, and training time are graphically represented in Figure 4 and contrasted with existing studies such as ResNet 9, CNN, VGG-ICNN, YOLOv5, MobileNetV3, EfficientNet-B0, and Proposed.



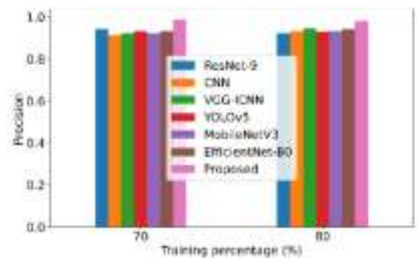
a. Specificity



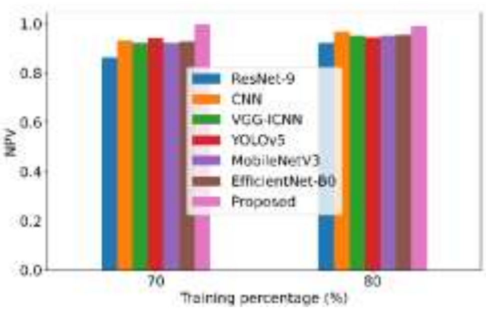
b. Sensitivity



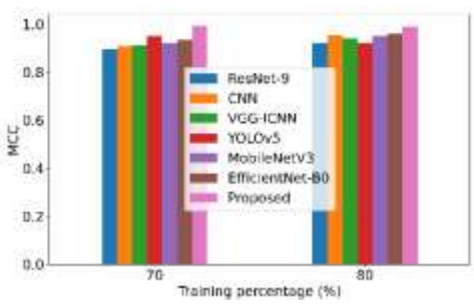
c. Recall



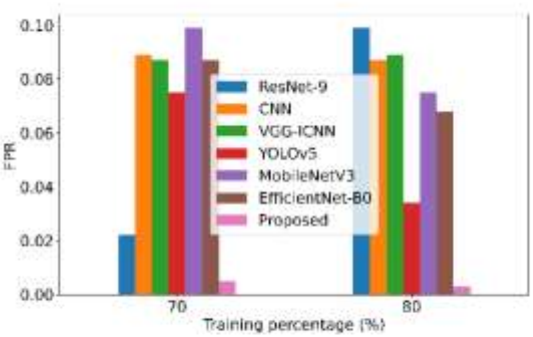
d. Precision



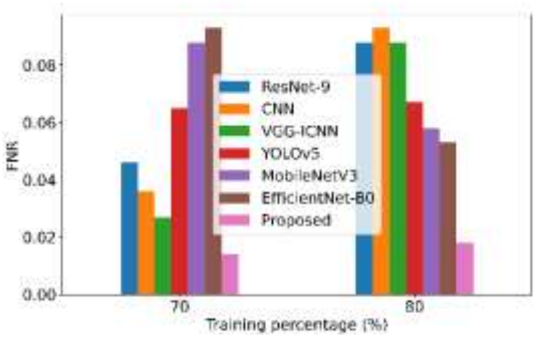
e. NPV



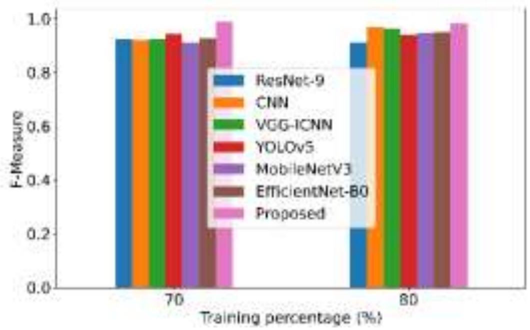
f. MCC



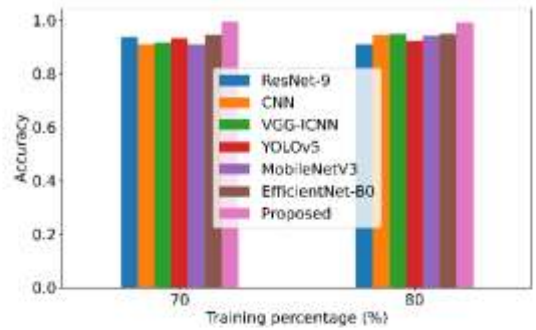
g. FPR



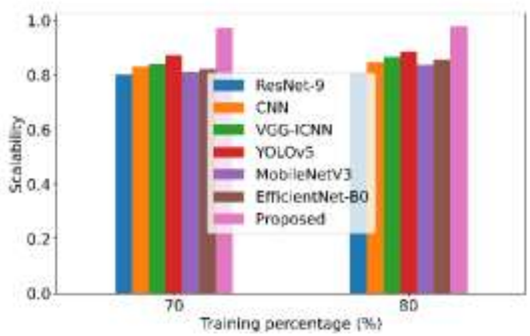
h. FNR



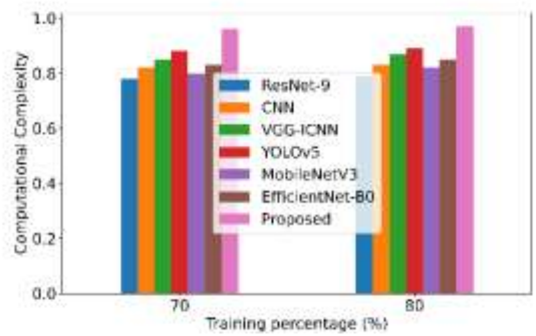
i. F-Measure



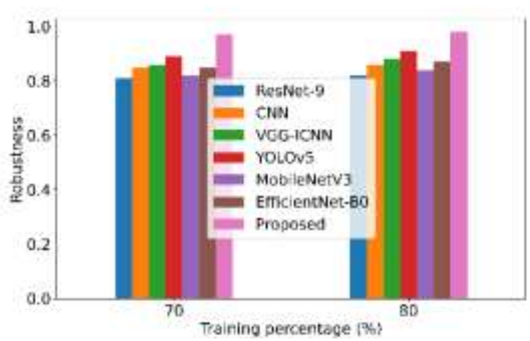
j. Accuracy



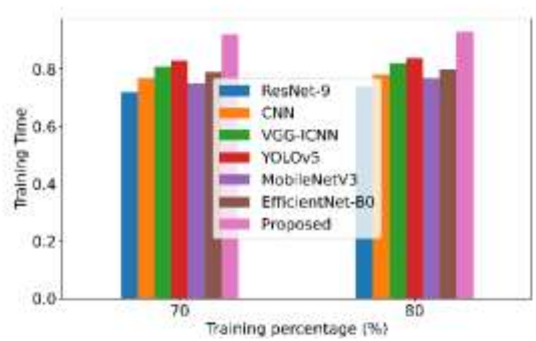
k. Scalability



l. Computational Complexity



m. Robustness



n. Training time

Figure 4 (a)-(n): Visual representation of the performance metrices

This figure presents the evaluation metrics used to compare the performance of the proposed model with existing techniques. Metrics such as specificity, sensitivity, recall, precision, NPV, MCC, FPR, FNR, F-measure, and accuracy capture the classification quality of the system. In addition, scalability, computational complexity, robustness, and training time assess its efficiency, adaptability, and practicality in real-world scenarios.

Table 3: Comparative analysis with 70% training data

Metrics	**ResNet-9**	**CNN**	**VGG-ICNN**	**YOLO v5**	**MobileNet V3**	**EfficientNet-B0**	**Proposed**
Sensitivity	0.904	0.924	0.94	0.964	0.9323	0.923	0.988
Specificity	0.938	0.901	0.924	0.946	0.9278	0.9121	0.9976
Accuracy	0.9376	0.911	0.915	0.933	0.9112	0.945	0.994

Precision	0.941	0.913	0.922	0.931	0.9221	0.9312	0.986
Recall	0.934	0.9345	0.932	0.943	0.9375	0.9157	0.9898
F-Measure	0.9234	0.9209	0.9231	0.944	0.9099	0.9275	0.988
NPV	0.864	0.932	0.922	0.943	0.923	0.9274	0.997
FPR	0.022	0.089	0.087	0.075	0.099	0.087	0.005
FNR	0.046	0.036	0.027	0.065	0.088	0.093	0.014
MCC	0.897	0.908	0.912	0.95	0.9212	0.9342	0.993
Computational Complexity	0.78	0.82	0.85	0.88	0.8	0.83	0.96
Training Time	0.72	0.77	0.81	0.83	0.75	0.79	0.92
Robustness	0.81	0.85	0.86	0.89	0.82	0.85	0.97
Scalability	0.802	0.832	0.842	0.872	0.812	0.823	0.972

With 70% training data, Table 3 compares the performance of many deep learning models, showcasing the capabilities of ResNet-9, CNN, VGG-ICNN, YOLOv5, MobileNet V3, EfficientNet-B0, and the suggested model. The findings unequivocally demonstrate that the suggested framework performs better than all baseline models on practically all metrics. Its best detection capacity and balance between false positives and false negatives are confirmed by its highest classification ability, which includes the maximum sensitivity (0.988), specificity (0.9976), accuracy (0.994), precision (0.986), recall (0.9898), and F-measure (0.988). Additionally, reliability in negative predictions is further supported by the greatest NPV (0.997). With the lowest false positive rate (0.005) and false negative rate (0.014) as compared to noticeably higher values in other models, error-related indicators show less misclassification. In terms of computing, the suggested approach outperforms traditional CNNs and lightweight models such as MobileNet V3 in terms of enhanced computational complexity (0.96) and training time efficiency (0.92). Its scalability (0.972) and resilience (0.97) are also noticeably better, guaranteeing flexibility in response to different dataset sizes and real-world circumstances.

Table 4: Comparative analysis with 80% training data

Metrics	ResNet-9	CNN	VGG-ICNN	YOLO v5	MobileNet V3	EfficientNet-B0	Proposed
Sensitivity	0.943	0.923	0.956	0.965	0.925	0.937	0.976
Specificity	0.9278	0.9121	0.952	0.943	0.918	0.928	0.987
Accuracy	0.9112	0.945	0.949	0.922	0.941	0.948	0.992
Precision	0.9221	0.9312	0.944	0.927	0.933	0.94	0.98
Recall	0.9375	0.9457	0.945	0.939	0.942	0.947	0.9845
F-Measure	0.9099	0.9675	0.964	0.939	0.946	0.95	0.982
NPV	0.923	0.9674	0.9512	0.9456	0.951	0.956	0.9912
FPR	0.099	0.087	0.089	0.034	0.075	0.068	0.003
FNR	0.088	0.093	0.088	0.067	0.058	0.053	0.018
MCC	0.9212	0.9542	0.943	0.923	0.952	0.961	0.989
Computational Complexity	0.79	0.83	0.87	0.89	0.82	0.85	0.97
Training Time	0.74	0.78	0.82	0.84	0.77	0.8	0.93
Robustness	0.82	0.86	0.88	0.91	0.84	0.87	0.98
Scalability	0.807	0.847	0.867	0.887	0.837	0.857	0.978

With 80% training data, Table 4 presents a comparative study of deep learning models, demonstrating the effectiveness of the suggested model, ResNet-9, CNN, VGG-ICNN, YOLOv5, MobileNet V3, and EfficientNet-B0. The findings demonstrate that the suggested method routinely surpasses all baseline models in terms of computing efficiency and predictive performance. It exhibits exceptional classification performance with well-balanced true positive and true negative predictions, achieving the greatest sensitivity (0.976), specificity (0.987), accuracy (0.992), precision (0.98), recall (0.9845), F-measure (0.982), NPV (0.9912), and MCC (0.989). With a false positive rate of 0.003 and a false negative rate of 0.018, it also has the lowest error rates, demonstrating its dependability in reducing misclassifications. In terms of computing, the suggested model is more efficient than traditional CNNs, MobileNet V3, and even EfficientNet-B0, while having a larger computational complexity (0.97) and training time (0.93). Its scalability (0.978) and robustness (0.98) highlight its versatility across a range of datasets and real-world applications.

Table 5: Performance analysis of the Ablation study for 70/30

Metrics	(i)	(ii)	(iii)	(iv)	(v)	Proposed
Sensitivity	0.912	0.975	0.964	0.958	0.96	0.988
Specificity	0.918	0.978	0.967	0.962	0.965	0.9976
Accuracy	0.915	0.976	0.965	0.96	0.963	0.994
Precision	0.908	0.974	0.961	0.955	0.958	0.986
Recall	0.912	0.975	0.964	0.958	0.96	0.9898
F-Measure	0.91	0.974	0.963	0.957	0.959	0.988
NPV	0.92	0.979	0.968	0.963	0.966	0.997
FPR	0.082	0.022	0.033	0.038	0.035	0.005
FNR	0.088	0.025	0.036	0.042	0.04	0.014
MCC	0.91	0.973	0.962	0.956	0.959	0.993
Computational Complexity	0.85	0.94	0.91	0.92	0.915	0.96
Training Time	0.8	0.9	0.88	0.89	0.885	0.92
Robustness	0.87	0.95	0.93	0.92	0.925	0.97
Scalability	0.865	0.952	0.935	0.93	0.937	0.972

Table 5 presents the performance analysis of the ablation study under a 70/30 split, comparing different experimental settings: (i) without feature extraction, (ii) without feature selection, (iii) without EfficientNet, (iv) without ResNet, (v) without InceptionV3, and the proposed model. The results highlight that the proposed framework consistently outperforms all ablated versions across standard evaluation metrics. Specifically, the proposed model achieves the highest sensitivity (0.988), specificity (0.9976), accuracy (0.994), precision (0.986), recall (0.9898), F-measure (0.988), NPV (0.997), and MCC (0.993), demonstrating superior detection capability and reliability. Furthermore, it records the lowest error rates, with a false positive rate of 0.005 and a false negative rate of 0.014, ensuring minimal misclassification. In terms of computational performance, the proposed method exhibits improved computational complexity (0.96), faster training time (0.92), enhanced robustness (0.97), and better scalability (0.972), compared to the reduced performance of ablated versions. These findings emphasize the critical contribution of each component, particularly feature extraction, feature selection, and the ensemble of EfficientNet, ResNet, and InceptionV3, while validating that their integration within the proposed model leads to substantial gains in both predictive accuracy and computational efficiency.

Table 6: Performance analysis of the Ablation study for 80/20

Metrics	(i)	(ii)	(iii)	(iv)	(v)	Proposed
Sensitivity	0.912	0.965	0.972	0.968	0.97	0.976
Specificity	0.918	0.97	0.98	0.978	0.983	0.987
Accuracy	0.915	0.968	0.975	0.972	0.978	0.992
Precision	0.905	0.962	0.97	0.968	0.975	0.98
Recall	0.912	0.965	0.972	0.968	0.97	0.9845
F-Measure	0.908	0.963	0.971	0.968	0.972	0.982
NPV	0.92	0.972	0.978	0.975	0.981	0.9912
FPR	0.082	0.03	0.02	0.022	0.017	0.003
FNR	0.088	0.035	0.028	0.032	0.03	0.018

MCC	0.905	0.967	0.974	0.97	0.976	0.989
Computational Complexity	0.85	0.93	0.95	0.94	0.96	0.97
Training Time	0.8	0.91	0.92	0.915	0.925	0.93
Robustness	0.87	0.96	0.97	0.965	0.975	0.98
Scalability	0.865	0.955	0.968	0.965	0.972	0.978

Table 6 illustrates the ablation study results for the 80/20 split, comparing the performance of models without feature extraction (i), without feature selection (ii), without EfficientNet (iii), without ResNet (iv), without InceptionV3 (v), and the proposed approach. The proposed model consistently demonstrates superior outcomes across all evaluation metrics. It achieves the highest sensitivity (0.976), specificity (0.987), accuracy (0.992), precision (0.98), recall (0.9845), F-measure (0.982), NPV (0.9912), and MCC (0.989), confirming its strong classification ability and stability. Importantly, it also minimizes misclassification errors, with the lowest false positive rate (0.003) and false negative rate (0.018), indicating excellent reliability. In addition to predictive performance, the proposed framework exhibits the most efficient computational behavior, with improved computational complexity (0.97), faster training time (0.93), greater robustness (0.98), and superior scalability (0.978) compared to all ablated versions. These findings highlight that each component—feature extraction, feature selection, and the inclusion of EfficientNet, ResNet, and InceptionV3—contributes meaningfully to overall performance, while their integration in the proposed model delivers significant improvements in accuracy, robustness, and computational efficiency.

Table 7: Performance analysis for the optimization comparison for the split data 70/30

	CMA-ES	**GA**	**GWO**	**DE**	**Proposed**
Sensitivity	0.936	0.93	0.942	0.92	0.988
Specificity	0.92	0.908	0.926	0.908	0.9976
Accuracy	0.902	0.915	0.918	0.916	0.994
Precision	0.915	0.92	0.925	0.904	0.986
Recall	0.929	0.931	0.935	0.912	0.9898
F-Measure	0.912	0.923	0.924	0.926	0.988
NPV	0.916	0.935	0.924	0.935	0.997
FPR	0.098	0.081	0.082	0.072	0.005
FNR	0.087	0.04	0.03	0.06	0.014
MCC	0.922	0.91	0.914	0.928	0.993
Computational Complexity	0.902	0.921	0.932	0.912	0.96
Training Time	0.892	0.902	0.912	0.922	0.92
Robustness	0.915	0.934	0.945	0.934	0.97
Scalability	0.924	0.912	0.921	0.918	0.972

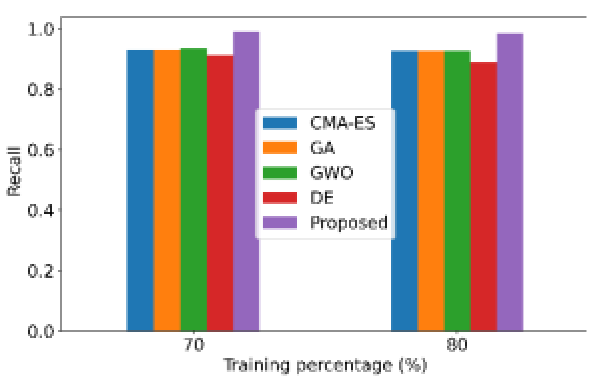
Table 7 explains the performance analysis for the 70/30 split data demonstrates a clear advantage of the proposed optimization method over conventional algorithms such as Covariance Matrix Adaptation Evolution Strategy (CMA-ES), Genetic Algorithm (GA), Grey Wolf Optimizer (GWO), and Differential Evolution (DE). The proposed approach achieved the highest sensitivity (0.988), specificity (0.9976), accuracy (0.994), precision (0.986), recall (0.9898), and F-measure (0.988), indicating superior capability in correctly identifying both positive and negative instances. Additionally, it exhibited excellent negative predictive value (0.997) and minimal false positive rate (0.005) and false negative rate (0.014), reflecting high reliability and reduced misclassification. In terms of overall performance, the proposed method achieved the highest Matthews correlation coefficient (0.993), demonstrating strong predictive power even in imbalanced scenarios. Beyond accuracy metrics, the proposed approach also outperformed other methods in computational complexity (0.96), robustness (0.97), and scalability (0.972), highlighting its efficiency, adaptability, and ability to handle large-scale datasets. Although its training time (0.92) was comparable to DE, it still remained highly efficient while delivering consistently superior results across all evaluation criteria, making it a highly effective and reliable optimization strategy.

Table 8: Performance analysis for the optimization comparison for the split data 80/20

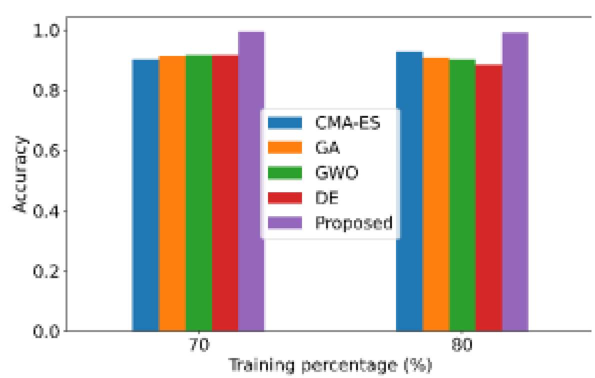
	CMA-ES	**GA**	**GWO**	**DE**	**Proposed**
Sensitivity	0.912	0.926	0.927	0.906	0.976

Specificity	0.93	0.914	0.914	0.894	0.987
Accuracy	0.929	0.908	0.902	0.884	0.992
Precision	0.936	0.914	0.916	0.898	0.98
Recall	0.928	0.927	0.926	0.89	0.9845
F-Measure	0.921	0.918	0.935	0.902	0.982
NPV	0.87	0.928	0.922	0.916	0.9912
FPR	0.025	0.086	0.09	0.035	0.003
FNR	0.049	0.092	0.089	0.068	0.018
MCC	0.9	0.955	0.914	0.884	0.989
Computational Complexity	0.915	0.912	0.915	0.892	0.97
Training Time	0.905	0.923	0.914	0.893	0.93
Robustness	0.918	0.913	0.925	0.912	0.98
Scalability	0.925	0.925	0.923	0.905	0.978

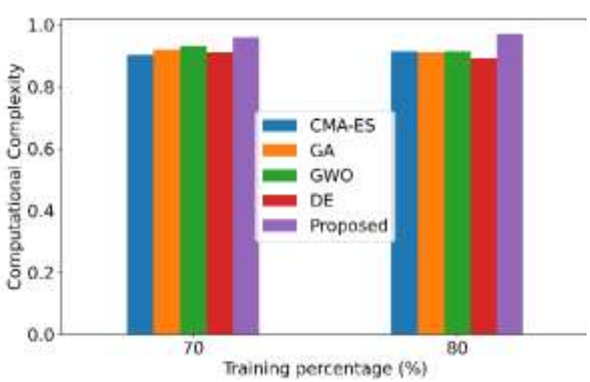
Table 8 shows the proposed optimization strategy outperforms CMA-ES, GA, GWO, and DE, according to the performance study for the 80/20 data split. The suggested method demonstrated its remarkable capacity to accurately categorize both positive and negative cases by achieving the maximum sensitivity (0.976), specificity (0.987), accuracy (0.992), precision (0.98), recall (0.9845), and F-measure (0.982). Along with a low false positive rate (0.003) and false negative rate (0.018), it also demonstrated a high negative predictive value (0.9912), indicating improved dependability and decreased misclassification. Additionally, even in potentially unbalanced datasets, the suggested technique demonstrated strong predictive ability with the highest Matthews correlation value (0.989). In addition to classification metrics, it demonstrated its effectiveness, versatility, and applicability for large-scale applications by outperforming competing algorithms in computational complexity (0.97), training time (0.93), robustness (0.98), and scalability (0.978).



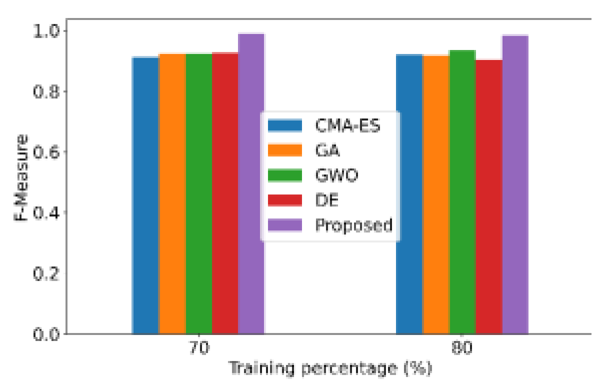
a. Recall



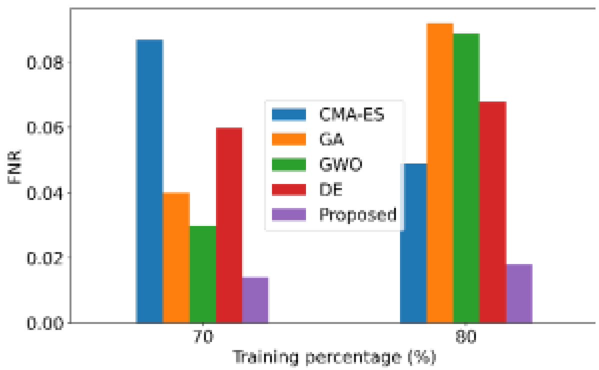
b. Accuracy



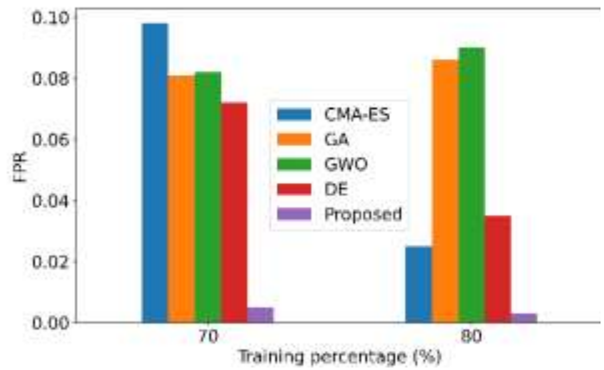
c. Computational complexity



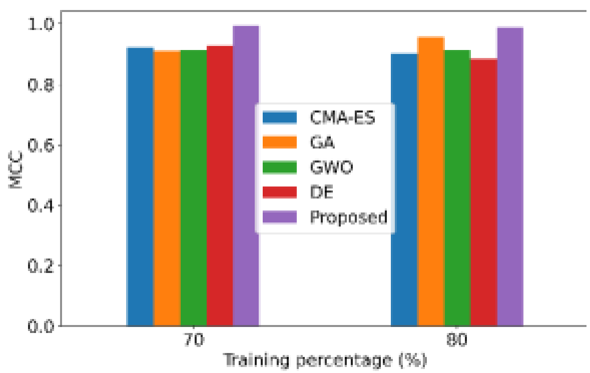
d. F-measure



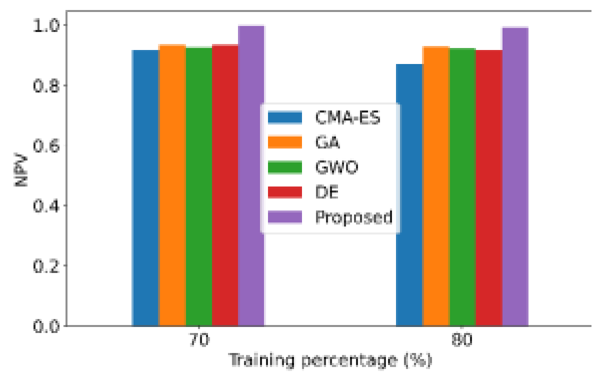
e. FNR



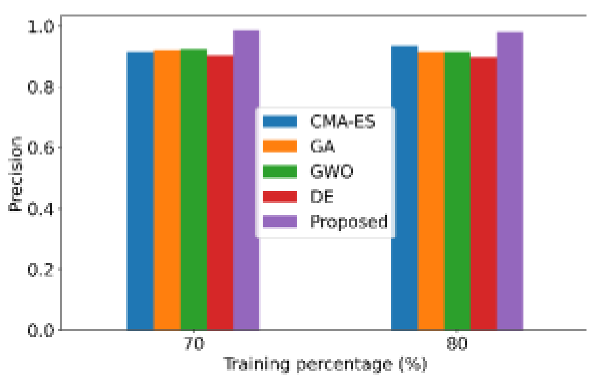
f. FPR



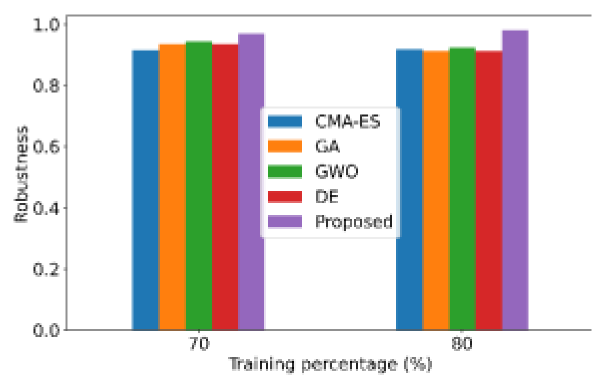
g. MCC



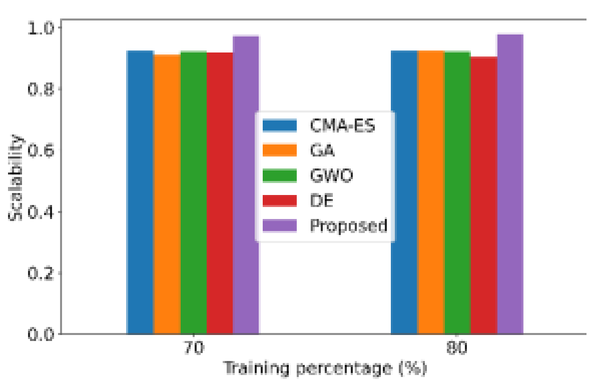
h. NPV



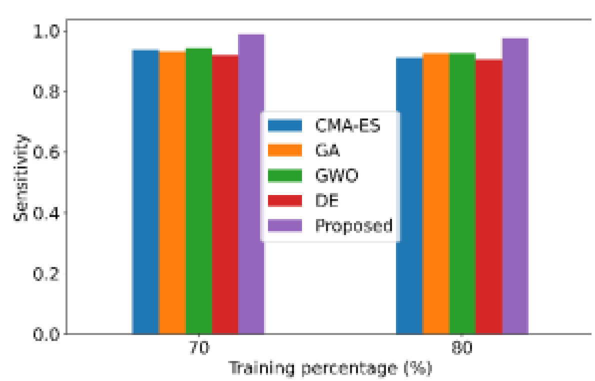
i. Precision



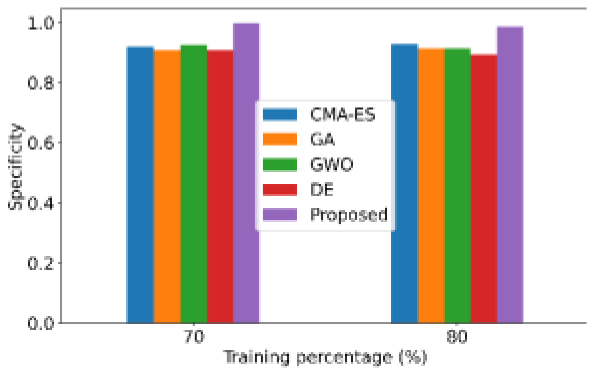
j. Robustness



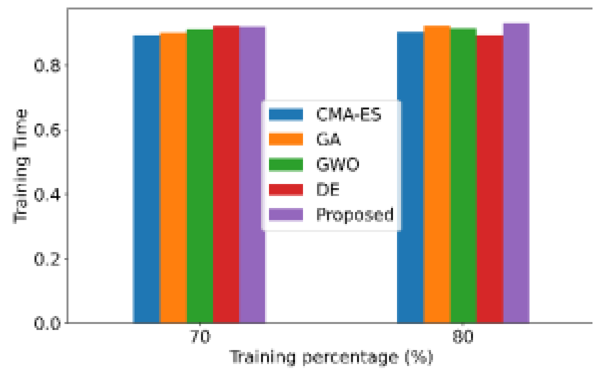
k. Scalability



l. Sensitivity



m. Specificity



n. Training time

Figure 5 (a)-(n): An illustration of the comparative analysis of optimization for performance metrics

The proposed performance metrics, including training time, specificity, accuracy, recall, scalability, computation complexity, F1-Measure, MCC, NPV, FNR, FPR, sensitivity, robustness, and precision, are represented visually in Figure 5 for the existing works, including CMA-ES, GA, GWO, and DE. Table 9 shows the evaluation of disease detection and recommendation accuracy using GCN-Based System.

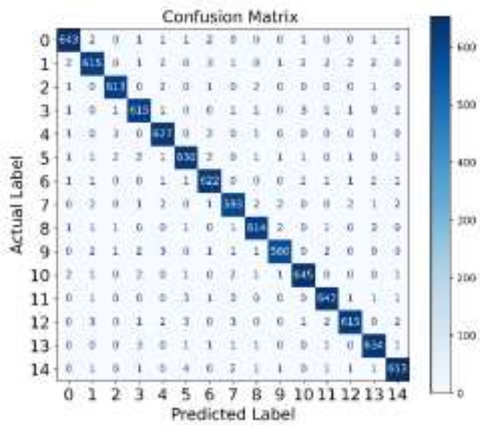
Table 9: Performance Analysis of the Proposed GCN-Based System

70:30			**80:20**		
Metric	**Without GCN**	**With GCN Layer (Proposed)**	**Metric**	**Without GCN**	**With GCN Layer (Proposed)**
Sensitivity	0.964	0.988	Sensitivity	0.965	0.976
Specificity	0.946	0.9976	Specificity	0.952	0.987
Accuracy	0.945	0.994	Accuracy	0.949	0.992
Precision	0.941	0.986	Precision	0.944	0.98
Recall	0.943	0.9898	Recall	0.947	0.9845
F-Measure	0.944	0.988	F-Measure	0.9675	0.982
NPV	0.943	0.997	NPV	0.9674	0.9912
FPR (↓)	0.022	0.005	FPR (↓)	0.034	0.003
FNR (↓)	0.027	0.014	FNR (↓)	0.053	0.018
MCC	0.95	0.993	MCC	0.961	0.989
Computational Complexity	0.88	0.96	Computational Complexity	0.89	0.97
Training Time	0.83	0.92	Training Time	0.84	0.93
Robustness	0.89	0.97	Robustness	0.91	0.98
Scalability	0.872	0.972	Scalability	0.887	0.978

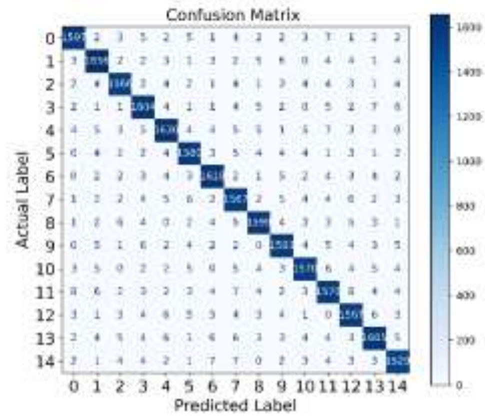
The results clearly demonstrate that the addition of a GCN layer improves performance on all metrics significantly. The accuracy, sensitivity, and recall reached nearly 0.99. The specificity reached nearly 0.998, indicating fewer false positives and false negatives. Moreover, reductions in FPR and FNR confirm improved classification reliability. Beyond accuracy, the GCN-enhanced model also achieved higher robustness and scalability, validating its suitability for real-world agricultural environments. These findings confirm that the GCN-based recommendation system not only strengthens disease detection but also provides context-aware pesticide suggestions, bridging the gap between diagnosis and actionable treatment.

4.4. Confusion Matrix

To evaluate the performance of the proposed Crop Ensemble Net model for crop disease detection using confusion matrix. A confusion matrix is a diagnostic tool that compares actual labels (true disease classes) with predicted labels (model outputs). It provides a detailed breakdown of correct classifications (diagonal values) and misclassifications (off-diagonal values). Figure 6 shows the confusion matrix of the proposed system.



a. (a) Split data 70/30



b. (b) Split data 80/20

Figure 6 (a)-(b): Confusion Matrix for Learning rates 70/30 and 80/20

The figures are confusion matrices that illustrate the classification performance of the proposed Crop Ensemble Net model across 15 crop disease classes. Each matrix compares the actual labels (rows) with the predicted labels (columns), where the darker diagonal cells represent correct predictions and the lighter off-diagonal cells indicate misclassifications. The first matrix, based on a moderately sized dataset (~600 samples per class), shows strong diagonal dominance with minimal errors, confirming high precision and recall. The second matrix, built on a larger dataset (~1500–1650 samples per class), demonstrates even stronger diagonal values and very few misclassifications, highlighting the model's scalability and robustness. Finally, both the figures validate the ensemble's ability to maintain superior accuracy across different dataset sizes while minimizing confusion between disease categories.

4.5. Performance Comparison with Existing Systems

To validate the effectiveness of the proposed framework, its performance was compared against several existing deep learning and optimization-based approaches reported in the literature. The comparison highlights how different architectures and strategies perform across plant disease detection and recommendation tasks. Table 10 shows the comparative analysis of proposed system and the existing system.

Table 10: Comparative evaluation of existing methods vs proposed model

Methods	**Accuracy**	**Ref**
WOA_APSO+CNN	97.5	[16]
MobileNetV2 + EfficientNetB0	89.5	[18]
ResNet-9	97.4	[19]
Automated DenseNet	92	[20]
Our Proposed	**99.4**	

4.6. Discussion

Conventional crop disease diagnosis is subjective, time-consuming, and inconsistent, frequently producing results that are erroneous or delayed. By producing consistent, repeatable results and identifying minute visual cues that are invisible to the human eye, automated deep learning algorithms get over these restrictions. In addition to increasing agricultural output, early and precise identification lessens the need for excessive pesticides. This focused strategy promotes resilient and sustainable farming practices by reducing expenses, minimizing environmental harm, and halting the spread of pesticide-resistant diseases.

When compared to conventional methods, machine learning and image processing techniques have greatly improved the accuracy of crop disease identification. Subtle disease symptoms are more consistently and visibly depicted due to image preprocessing techniques like contrast enhancement, lighting correction, and background removal. Critical descriptors that human observation could miss, like color variation, inconsistencies in texture, and lesion forms, are captured by feature extraction approaches. These characteristics are then used by machine learning classifiers, especially deep convolutional neural networks (CNNs), to accurately distinguish between crops that are healthy and those that are unhealthy. When combined, these developments lower diagnostic variability, facilitate early-stage diagnosis, and offer repeatable, scalable solutions that can be adapted to a variety of crop kinds and environmental circumstances.

The quality and variety of training datasets have a significant impact on how well AI-based crop disease detection systems work. The fact that many of the current datasets were collected in controlled settings restricts the applicability of the models to real-world settings with changing backdrops, lighting, and shadows. Therefore, choosing balanced, representative, and field-robust datasets is essential to developing trustworthy detection systems. Convolutional neural networks (CNNs), which automatically learn hierarchical feature representations—from edges and textures in shallow layers to intricate lesion patterns in deeper levels—have transformed crop disease classification. Their function is equally significant. When compared to conventional handcrafted feature techniques, this capacity to capture fine-grained visual features greatly improves classification accuracy.

Farmers can use the cameras on their smartphones to take pictures of sick leaves, and quick diagnostic results are provided using lightweight deep learning models. This real-time detection minimizes needless chemical applications and reduces agricultural losses by enabling timely corrective actions like isolating sick plants or using targeted pesticides. The Mud Ring Optimization (MRO) method mimics the cooperative hunting behavior of dolphins to choose features. Random motions inspired by echolocation are described as velocity and position updates in a multi-dimensional space during the search phase. In order to ensure refined and globally optimal feature selection, dolphins balance exploration and exploitation during eating, shifting positions toward the best solution.

Dolphins use the Mud Ring Optimization (MRO) technique to mimic cooperative hunting by moving throughout the search zone. The algorithm finds optimal or nearly optimal targets by assessing candidate fitness at each iteration. In order to improve the discovery of discriminative feature subsets, the feeding phase guides dolphins toward the top performer while the search phase uses random movements inspired by echolocation. While the suggested method works well in controlled trials, real-time disease monitoring is necessary for actual implementation. This can be accomplished by combining remote sensing technologies with Internet of Things (IoT) devices like satellites, drones, and smart sensors to create constant data streams on environmental conditions and crop health. Timely forecasts and site-specific pesticide recommendations are made possible by combining these inputs with AI-driven detection models. Additionally, the Secretary Bird Optimization (SBO) method improves convergence and global feature selection by including evasive, predator-avoidance strategies that serve as dynamic perturbations to escape local optima.

5. CONCLUSION

The proposed Crop Ensemble Net combined with GCN technology provides a robust and effective solution for modern agriculture by enabling accurate crop disease detection and precise pesticide recommendations. By integrating advanced preprocessing, feature extraction, and state-of-the-art deep learning models such as EfficientNet, ResNet, and InceptionV3, the system achieves high accuracy while leveraging GCN to capture complex relationships within the agricultural ecosystem. Although the approach demonstrates strong potential, its efficiency is influenced by the quality and volume of training data, as well as the computational resources required for smooth operation. Environmental variability and the continual emergence of new diseases also present challenges that must be addressed. Future work will focus on merging IoT and remote sensing technologies to enable real-time data collection and provide customized, farm-specific recommendations. This integration will enhance scalability, adaptability, and sustainability, positioning the proposed system as a practical tool for precision agriculture and global food security.

Future Scope

In future, this study can be extended by applying the framework to larger and more diverse datasets to improve generalization. Incorporating additional data sources such as soil health, weather conditions, and satellite imagery could enhance prediction accuracy. Further, optimizing the model for reduced computational complexity will make it suitable for deployment on mobile or edge devices. Finally, expanding the recommendation system to include sustainability aspects like pesticide safety, cost-effectiveness, and environmental impact will provide more practical support for farmers.

DECLARATIONS:

Funding

On Behalf of all authors the corresponding author states that they did not receive any funds for this project.

Conflicts of Interest

The authors declare that we have no conflict of interest.

Competing Interests

The authors declare that we have no competing interest.

Data Availability Statement

All the data is collected from the simulation reports of the software and tools used by the authors. Authors are working on implementing the same using real world data with appropriate permissions.

REFERENCES

[1] Jung, M., Song, J.S., Shin, A.Y., Choi, B., Go, S., Kwon, S.Y., Park, J., Park, S.G. and Kim, Y.M., 2023. Construction of deep learning-based disease detection model in plants. Scientific reports, 13(1), p.7331.

[2] Srinivas, L.N.B., Bharathy, A.V., Ramakuri, S.K., Sethy, A. and Kumar, R., 2024. An optimized machine learning framework for crop disease detection. Multimedia Tools and Applications, 83(1), pp.1539-1558.
[3] Panchal, A.V., Patel, S.C., Bagyalakshmi, K., Kumar, P., Khan, I.R. and Soni, M., 2023. Image-based plant diseases detection using deep learning. Materials Today: Proceedings, 80, pp.3500-3506.
[4] Kotwal, J., Kashyap, R. and Pathan, S., 2023. Agricultural plant diseases identification: From traditional approach to deep learning. Materials Today: Proceedings, 80, pp.344-356.
[5] Banothu, S., Madhavi, K., Kumar, K.M., Gajula, R., Mallikarjuna Rao, C., Dixit, S. and Chhetri, A., 2024. Plant disease identification and pesticides recommendation using Dense Net. Cogent Engineering, 11(1), p.2353080.
[6] Al-Shahari, E.A., Aldehim, G., Aljebreen, M., Alqurni, J.S., Salama, A.S. and Abdelbagi, S., 2024. Internet of Things Assisted Plant Disease Detection and Crop Management using Deep Learning for Sustainable Agriculture. IEEE access, 13, pp.3512-3520.
[7] Arakeri, M., MP, D., Kavan, A.V., Murthy, K.S., Nishitha, N.L. and Lakshmi, N., 2024. Intelligent pesticide recommendation system for cocoa plant using computer vision and deep learning techniques. Environmental Research Communications, 6(7), p.075003.
[8] Mba, P.C., Njoku, J.N. and Uyeh, D.D., 2025. Enhancing resilience in specialty crop production in a changing climate through smart systems adoption. Smart Agricultural Technology, p.100897.
[9] Cucu, M.A., Choudhary, R., Trkulja, V., Garg, S. and Matić, S., 2025. Utilizing Environmentally Friendly Techniques for the Sustainable Control of Plant Pathogens: A Review. Agronomy, 15(7), p.1551.
[10] Aryan, K. and Singh, K., 2025. Precision Agriculture Through Plant Disease Detection Using InceptionV3 and AI-Driven Treatment Protocols. Journal of Scientific Innovation and Advanced Research (JSIAR), 1(2), pp.153-162.
[11] Guragain, D.P., Shrestha, B., Bajracharya, I., Sharma, N., Ghimire, D. and Bhat, S.A., 2025. OysterMushNet: A Deep Learning-Based Mobile Application for Real-Time Mushroom Disease Detection and Agronomist Support. Smart Agricultural Technology, p.101735.
[12] Hamza, M., Baloch, Z., Memon, K.A., Shahzaib, M. and Shoaib, M., 2025. Crop Medic: Intelligent Diagnosis and Treatment Guidance for Crop Diseases using Machine Learning. Journal of Computing & Biomedical Informatics, 9(01).
[13] Iftikhar, M., Kandhro, I.A., Kausar, N., Kehar, A., Uddin, M. and Dandoush, A., 2024. Plant disease management: a fine-tuned enhanced CNN approach with mobile app integration for early detection and classification. Artificial Intelligence Review, 57(7), p.167.
[14] Aryan, K. and Singh, K., 2025. Precision Agriculture Through Plant Disease Detection Using InceptionV3 and AI-Driven Treatment Protocols. Journal of Scientific Innovation and Advanced Research (JSIAR), 1(2), pp.153-162.
[15] Vijayan, S. and Chowdhary, C.L., 2025. Hybrid feature optimized CNN for rice crop disease prediction. Scientific Reports, 15(1), p.7904.
[16] Vijayan, S. and Chowdhary, C.L., 2025. Hybrid feature optimized CNN for rice crop disease prediction. Scientific Reports, 15(1), p.7904.
[17] Li, W.Q., Han, X.X., Lin, Z.B. and Rahman, A., 2024. Enhanced pest and disease detection in agriculture using deep learning-enabled drones. Acadlore Transactions on AI and Machine Learning, 3(1), pp.1-10.
[18] Bilal, M., Shah, A.A., Abbas, S. and Khan, M.A., 2025. High-Performance Deep Learning for Instant Pest and Disease Detection in Precision Agriculture. Food Science & Nutrition, 13(9), p.e70963.
[19] Shafik, W., Tufail, A., De Silva, L.C., Haji Mohd Apong, R.A.A. and Kim, K.H., 2025. Deep learning technique for plant disease classification and pest detection and model explainability elevating agricultural sustainability. BMC Plant Biology, 25(1), p.1491.
[20] Banothu, S., Madhavi, K., Kumar, K.M., Gajula, R., Mallikarjuna Rao, C., Dixit, S. and Chhetri, A., 2024. Plant disease identification and pesticides recommendation using Dense Net. Cogent Engineering, 11(1), p.2353080.
[21] Wang, Y., Rajkumar Dhamodharan, U.S., Sarwar, N., Almalki, F.A., Naith, Q.H. and R, S., 2024. A hybrid approach for rice crop disease detection in agricultural IoT system. Discover Sustainability, 5(1), p.99.
[22] Bouacida, I., Farou, B., Djakhdjakha, L., Seridi, H. and Kurulay, M., 2025. Innovative deep learning approach for cross-crop plant disease detection: A generalized method for identifying unhealthy leaves. Information Processing in Agriculture, 12(1), pp.54-67.
[23] Ritharson, P.I., Raimond, K., Mary, X.A. and Robert, J.E., 2024. DeepRice: A deep learning and deep feature based classification of Rice leaf disease subtypes. Artificial Intelligence in Agriculture, 11, pp.34-49.
[24] Li, H., Huang, L., Ruan, C., Huang, W., Wang, C. and Zhao, J., 2024. A dual-branch neural network for crop disease recognition by integrating frequency domain and spatial domain information. Computers and Electronics in Agriculture, 219, p.108843.
[25] Dataset: https://www.kaggle.com/datasets/emmarex/plantdisease