Paddy Disease Recognition and Classification using Artificial Neural Networks: Introducing a Novel Hybrid Optimization Algorithm

Farida Siddiqi Prity

MASTER OF SCIENCE IN INFORMATION AND COMMUNICATION ENGINEERING

NOAKHALI SCIENCE AND TECHNOLOGY UNIVERSITY

Thesis Report

Paddy Disease Recognition and Classification using Artificial Neural Networks: Introducing a Novel Hybrid Optimization Algorithm

Submitted By

Farida Siddiqi Prity

Roll: BKH2111MSc101F

Session: 2020-21

A Thesis Report
Submitted in Partial Fulfillment of the Requirements
for the Degree of Master of Science (Engg.)
in the Department of Information and communication Engineering

Thesis Supervisor

Dr. Md. Ashikur Rahman Khan

Chairman and Professor

Department of Information & Communication Engineering

Noakhali Science & Technology University

Noakhali- 3814

Bangladesh

20 March, 2024

DECLARATION

This thesis report is submitted to the Department of Information and Communication Engineering (ICE), Noakhali Science and Technology University, Noakhali- 3814, in order to fulfillment of the requirements for having the M.Sc. degree in ICE under the course entitled "ICE- 5312". So, I, hereby declare that this thesis report has not been submitted elsewhere for the requirement of any kind of degree, diploma or publication.

Farida Siddiqi Prity
Roll No: BKH2111MSc101F
Session: 2020-21
Year: 2, Term: 1
Department of Information and Communication Engineering
Noakhali Science and Technology University
Noakhali- 3814.

ACCEPTANCE

This thesis report is submitted to the Department of Information and Communication Engineering (ICE), Noakhali Science and Technology University, Noakhali- 3814, in order to partial fulfillment of the requirements for having the M.Sc. degree in ICE under the course entitled “ICE- 5312”.

Dr. Md. Ashikur Rahman Khan
Chairman and Professor
Department of Information and Communication Engineering
Noakhali Science and Technology University
Noakhali- 3814.

ABSTRACT

Paddy diseases have significantly deteriorated the production capacity of the paddy plant, resulting in substantial economic losses. It becomes imperative to identify paddy diseases early to effectively mitigate and minimize the detrimental impacts of these diseases. Timely detection of infections enables effective management strategies and leads to enhanced yield in terms of quantity and quality. Hence, it is imperious to develop a methodology that can swiftly and accurately detect paddy diseases. Artificial Neural Networks (ANNs) have emerged as a promising approach for paddy disease recognition. Nevertheless, training ANNs pose significant challenges, particularly when dealing with intricate networks containing numerous parameters. Prior research has explored single optimization algorithms for ANN training; however, these algorithms are hindered by constraints such as convergence problems, limited exploration capabilities, inadequate adaptability, and inconsistent performance across diverse problem domains. Consequently, there is a pressing demand for a hybrid optimization algorithm that can effectively address these challenges by offering an optimal solution for optimizing ANN parameters. Therefore, this paper aims to devise a novel hybrid optimization algorithm, namely Adaptive Particle Swarm Water Wave Optimization (APSWWO), for finding the disease in the paddy plant, where the training of ANN is completed using the APSWWO, which is formed by assimilating two nature-inspired optimization algorithms Adaptive Particle Swarm Optimization (APSO) and Water Wave Optimization (WWO). Initially, the images of paddy leaf disease are pre-processed and balanced. Two customized ANN models: 26-layer Convolutional Neural Network model and 13-layer Lightweight Vision Transformer (LiteViT) are used to classify ten types of paddy diseases and one healthy class. The optimization of parameters in the training process is achieved by incorporating the APSWWO algorithm. This study also employs Explainable AI techniques such as Gradient-weighted Class Activation Mapping (Grad-CAM) to provide model interpretability and visualizations in each layer, enabling more transparent and trustable disease detection. Accuracy, recall, precision, and F-measure metrics are utilized to assess classification performance. The hybrid approach's effectiveness is evaluated by comparing ANN techniques performance with and without optimization (Original, PSO, APSO, WWO, and APSWWO), highlighting the importance of hybrid APSWWO in enhancing prediction performance. The hybrid CNN_APSWWO outperforms other forms (CNN_Original, CNN_PSO, CNN_APSO, CNN_WWO, LiteViT_Original, LiteViT_PSO, LiteViT_APSO, LiteViT_WWO, and LiteViT_APSWWO), offering valuable insights for optimizing paddy disease recognition and classification, ultimately enhancing agricultural practices and contributing to improved crop health and food security.

Keywords: Paddy; Disease; Artificial Neural Network; Nature-inspired Optimization; Hybrid Algorithm; Explainable AI

TABLE OF CONTENT

LIST OF FIGURE

LIST OF TABLE

ABBREVIATION

AI	Artificial Intelligence
ANN	Artificial Neural Network
PSO	Particle Swarm Optimization
APSO	Adaptive Particle Swarm Optimization
WWO	Water Wave Optimization
APSWWO	Adaptive Particle Swarm Water Wave Optimization
CNN	Convolutional Neural Network
LiteViT	Light Weight Vision Transformer
GA	Genetic Algorithm
BFO	Biological Foraging Optimization
DWWO	Discrete Water Wave Optimization algorithm
NWFSP	No-Wait Flowshop Scheduling Problem
COP	Combinatorial Optimization Problem
CWWO	Cooperative Water Wave Optimization
MWWO	Modified Water Wave Optimization
GAN	Generative Adversarial Network
BB-BC	Big Bang–Big Crunch
SSA	Salp Swarm Algorithm
TP	True Positive
TN	True Negative
FP	False Positive
FN	False Negative

CHAPTER 1

INTRODUCTION

1.1 Background

Plant diseases are exerting a profound global impact on both food security and productivity. Globally, more than 800 million people grapple with food insufficiency, with almost 10% of global food production being adversely affected by plant diseases, yielding widespread ramifications [1]. This crisis disproportionately impacts more than 1.3 billion people living on less than $1 per day [2]. Moreover, it is crucial to emphasize that these plant ailments lead to substantial annual losses, resulting in estimated damages of $220 billion, equivalent to 10-16 percent of the global crop yield [3].

Paddy (Oryza *sativa*) is one of the most important staple crops worldwide, serving as a primary source of food for over half of the global population with a total consumption of 493.13 million metric tons in 2019–2020 and 486.62 in the year 2018–2019 [4]. This indicates a rise in paddy consumption compared to the metric tons consumed over the years. As per the FAO, a production increase of approximately 70% is required by 2050, equating to the delivery of 880 million tons of paddy to meet the growing population's needs [5]. However, paddy cultivation is constantly threatened by various diseases caused by pathogens such as fungi, bacteria, and viruses. If not promptly identified and treated, these diseases can cause significant yield losses, affecting regional economies, livelihoods, and food security in paddy-producing areas.

Paddy crop disease identification has always relied on visual examination by knowledgeable agronomists or plant pathologists. This method requires a lot of work, takes a long time, and is subjective, frequently resulting in a delayed and incorrect diagnosis. Additionally, there is an increasing need for automated, fast, and precise disease detection and identification techniques due to the expanding amount of rice consumption and the requirement for effective disease management.

Recent developments in computer vision and Artificial Intelligence (AI) technology have enormous potential for transforming how diseases are identified and controlled in agricultural systems. Artificial Neural Network algorithms have demonstrated excellent proficiency in image recognition and classification tasks [6-9]. These algorithms can learn complicated patterns and features from image data and make precise predictions using massive datasets and sophisticated computational resources. The ability to detect diseases early and accurately enables farmers to take prompt action, implement targeted interventions, and minimize the spread of pathogens, thereby preventing significant yield losses and reducing the reliance on agrochemicals. Although an ANN enables robust prediction accuracy and broad applicability, it is always challenging to accurately determine the network's hyperparameters and the connection weights (weights and bias). The relevant parameters are typically determined by manual parameter adjustment or empirical values in traditional research, which demands a large amount of labor and material resources and frequently results in an unsatisfactory parameter combination. An effective optimization strategy is needed to achieve the ideal ANN parameters. Farmers can more effectively detect rice disease early on with an optimized ANN approach.

This research aims to develop a model utilizing ANN that accurately detects paddy diseases, alongside the creation of a novel hybrid optimization algorithm, to efficiently optimize ANN parameters. Additionally, the study seeks to classify paddy leaf diseases using ANN, enabling farmers and agricultural professionals to take immediate action to preserve both the quantity and quality of paddy production through an early automatic detection system. Moreover, a comprehensive comparison among different neural network-based models and optimization algorithms will be conducted to identify the most effective approaches in disease detection and management.

1.1.1 Paddy Disease Description

Paddy has been an important staple meal for the economies of undeveloped countries and farmers, and the economies and farmers are heavily dependent upon the productivity of the paddy crop. According to FAO, it is developed on 166 Mha of land, with a paddy crop yield of 745.17 mt and a typical profit of about five t/ha [1]. According to estimates, 880 mt of hard rice, with an increase of almost 70%, must be produced by 2025 to meet the demand caused by the growing population [4].

Numerous diseases damage the paddy crop, which has a negative impact on yield quality and quantity. According to Gianessi et al. [10] and Singh et al. [11], these diseases can cause yield losses of 10%–30%. Furthermore, according to Liu et al. [12], yield losses might arise from disease incidence in rice plants ranging from 9.5% to 69%. Rice diseases can be detected by carefully observing symptoms that emerge on plant leaves, blooms, crop stems, or roots. Diseases of the rice plant primarily manifest as symptoms on the leaves. The wind, insects, water droplets from irrigation or rain, etc. spread these leaf ailments. Pathogens that harm rice crops leaves are Magnaporthe, Erwinia, Acidovorax, Burkholderia, Pseudomonas, Pantoea, rice tungro bacilliform virus (RTBV), rice tungro spherical virus (RTSV), Dickeya, and Enterobacter [11]. Table 1.1 represents the description of paddy leaf diseases. X. oryzae pv pathogens are responsible for the bacterial blight disease of paddy [10]. A paddy crop infected with blasts develops lesions on various plant components, which can harm the harvest and cause farmers to lose money. Magnaporthe oryzae is a fungus that causes rice blasts [13]. The bacterium Pseudomonas syringae pv. syringae is responsible for the brown spot disease of rice leaves [14]. Rice tungro disease, brought on by viruses spread by leafhoppers, causes sterility, stunted growth, and discoloration of the leaves [3]. The two viruses that cause tungro disease are rice tungro bacilliform virus (RTBV) and rice tungro spherical virus (RTSV) [15].

Table 1.1: Paddy disease description

Name of Disease	Description	Cause	Affected part of the plant	Damage
Bacterial Leaf Blight	Leaf lesions become water-soaked, turn brown, dry, and cause wilting and plant death.	Xanthomonas oryzae bacteria	Leaves	Wilting and plant death
Brown Spot	Brown spots, yellow borders, smaller grains	Cochliobolus miyabeanus fungus	Leaves, panicles, grains	Reduced yield and quality of rice
Hispa	Leaf holes on leaves	Dicladispa armigera insect	Leaves, leaf sheaths	Yield loss
Leaf Blast	Circular or elongated dead areas on leaves, panicles, and glumes, resulting in smaller and lower-quality grains.	Magnaporthe oryzae fungus	Leaves	Reduced yield and quality of rice
Leaf Scald	Lesions of yellow to brown color on leaf blades, leaf sheath, and leaf collar.	Xanthomonas oryzae pv. oryzicola bacterium	Leaves	Yield loss and plant death
Leaf Streak	Tan to brown circular or elongated leaf spots, decreased crop yield	Cercospora oryzae fungus	Leaves	Yield loss
Narrow Brown Spot	Reduced crop yield due to small, rectangular brown leaf spots.	Raphanus sativus var. nasturtii fungus	Leaves	Reduced yield
Sheath Blight	Reduced grain size and quality due to dark-brown to black lesions on leaf sheaths and stems.	Rhizoctonia solani fungus	Leaf sheath	Reduced yield
Tungro	Reduced grain size and quality, along with stunted growth, caused by chlorotic and necrotic leaf spots.	Rice tungro spherical virus and Rice tungro bacilliform virus	Leaves	Loss of yield

Accurate disease prediction and forecasting of paddy leaves are essential to preserve the quantity and quality of paddy production since detection at the initial stage of the disease helps ensure that timely intervention can be provided to convert the growth of the disease to facilitate the healthy growth of the plant for increasing production as well as the supply of rice. An efficient ANN technique can assist farmers in accurate disease prediction and forecasting of rice leaves. However, training ANNs can be challenging and computationally expensive, especially for complex networks with many parameters. This is where optimization algorithms play a crucial role in improving the training process and enhancing the performance of ANNs.

Different authors have used optimization techniques such as Bayesian optimization, Particle Swarm Optimization (PSO), and Big Bang-Big Crunch optimization algorithms in paddy disease detection [16]. But, it is proved that the hybrid optimization algorithm is more preferable to a single optimization algorithm due to its ability to enhance exploration and exploitation, improve robustness and adaptability, handle complex and nonlinear problems, perform well on different objective functions or constraints, and provide flexibility for customization [12]. By leveraging the strengths of diverse algorithms, hybrid optimization strikes a balance between exploration and exploitation, adapts to problem dynamics, and increases the chances of finding optimal or near-optimal solutions, making it a more effective approach for a wide range of optimization tasks. Although a few authors use single optimization techniques in paddy disease detection problems, the use of hybrid optimization algorithm is still lacking. So, there is a need for an efficient hybrid algorithm that will solve these problems by providing the optimal solution to effectively optimize the classifier's parameters and precisely detect rice disease.

1.2 Problem Statement

Paddy diseases severely hamper paddy production. Additionally, it poses a serious risk to food security. Therefore, it is crucial to diagnose and identify paddy diseases to maintain the high yield, high quality, and high efficiency of paddy. It is crucial to classify diseases accurately because different disease management strategies must be used depending on the pathogen, host, biotic, and abiotic components involved.

Large amounts of paddy were frequently destroyed due to disease-related issues due to a lack of cropland or improper monitoring. To combat plant diseases, agroecosystems have experienced adverse effects due to the widespread use of pesticides like bactericides, fungicides, and nematicides. According to Zarbafi et al. [13], Han et al. [14], and Sibin [15], the most common paddy diseases are sheath blight, bacterial blight, rice blast, and symptoms characterized by texture, color, and shape, which are typical of swift prevalence and easy infection. Currently, the process for detecting paddy diseases is anticipated to include artificial identification, mapping rice diseases, and automated detection. Paddy disease identification has traditionally been performed manually and has proven unreliable, expensive, and time-consuming. Although the mapping technique for disease diagnosis in rice is relatively simple to use, it is possible to misread some similar diseases that have detrimental effects on paddy growth. Due to the significant ambient effect, sluggish detection speed, and poor accuracy of the most recent computer-based identification method, it has not yet been widely adopted. Therefore, creating a method for detecting paddy diseases is crucial that can deliver prompt and precise judgments. ANN can provide an effective way to detect paddy disease. But, it is necessary to use optimization algorithms in the training process of ANN. Without optimization algorithms in ANN training, the convergence to optimal solutions would be slower or even unachievable, leading to suboptimal performance. The network may get trapped in local minima, hindering its ability to find the global minimum and diminishing predictive accuracy. The training process would be inefficient and computationally expensive, impeding scalability for complex networks with a large number of parameters. Some authors have used different optimization algorithms in ANN training. Despite some uses of individual optimization algorithms in optimizing the hypermeters of ANN in paddy disease detection, the use of hybrid algorithms is lacking. Hybrid algorithms can solve the problems of a single optimization algorithm, such as suffering from convergence to local optima, insufficient exploration, sensitivity to initialization or parameters, challenges in handling constraints or multi-objective optimization, limited adaptability and performance disparities across problem domains. So, there is a need for a hybrid optimization algorithm that will solve these problems by providing the optimal solution to optimize the parameters of the classifier effectively. Therefore, this study aims to propose a novel hybrid algorithm Adaptive Particle Swarm Water Wave Optimization (APSWWO) which is based on two optimization algorithms, Adaptive Particle Swarm Optimization (APSO) and Water Wave Optimization (WWO), to optimize the hyperparameters of ANN. This study has developed two customized ANN model: 26-layer CNN and 13-layer LiteViT to effectively classify paddy disease. The proposed novel hybrid APSWWO has been used to optimize the hyper-parameters of these two models. In addition, Grad-CAM was employed for interpretability in every layer of both models to gain insights into their decision-making processes and highlight areas of interest in the images related to paddy disease classification. The system's potential benefits extend beyond accurate disease diagnosis, enabling farmers to make data-driven decisions for targeted pesticide application, resource allocation, and crop management strategies. Ultimately, this research aims to contribute to the sustainable development of agriculture, enhance crop health management practices, and support global food security.

1.3 Research Gap

In recent years, paddy disease detection, utilizing ANNs and optimization techniques, has witnessed significant advancements. Numerous studies have contributed to developing ANN architectures for accurate disease identification and classification. While these efforts have undoubtedly expanded our understanding of the domain, several research gaps persist that necessitate further investigation and innovation. Various researchers have delved into the use of transfer learning in conjunction with CNNs for paddy disease detection. The datasets used in some studies are small and imbalanced, while others have employed large but imbalanced datasets. It is important to note that CNN experiments require substantial labeled data, and issues arise when dealing with unlabeled and imbalanced datasets. Furthermore, CNN training demands considerable computational resources. Consequently, the datasets employed by these studies may not be ideal for effective CNN and transfer learning training. Some studies employ transfer learning; however, this may introduce challenges related to domain adaptation and model generalization. Some studies have not incorporated image enhancement techniques, which is a notable research gap. Model interpretability, a crucial aspect for enhancing trust, transparency, and the decision-making process in machine learning models, has been overlooked by some studies. Further research gaps emerge concerning the optimization of ANN model parameters. Specifically, some studies did not employ any optimization algorithms to fine-tune their ANN model parameters, while others utilized single optimization techniques. While some studies have emphasized a single performance metric, focusing solely on accuracy may not provide a comprehensive evaluation of model performance. Moreover, even when achieving high accuracy, some studies did not employ any overfitting reduction methods. Additionally, some studies have primarily concentrated on the classification of 2-7 classes of paddy diseases, potentially limiting the practical utility of disease detection models.

1.4 Motivation of This Research

The motivation for researching paddy disease detection using Artificial Neural Networks lies in the need for efficient and accurate disease management in paddy cultivation. Paddy is a staple food for a significant portion of the global population, and diseases can cause substantial yield losses, impacting food security and the livelihoods of farmers. Traditional disease detection methods are often time-consuming, labor-intensive, and prone to errors. By leveraging the power of ANN-based models, the research aims to provide an automated, reliable, and timely solution for detecting and diagnosing rice diseases. Such a system can enable early intervention, effective disease management, and optimized use of resources, ultimately enhancing crop productivity, reducing losses, and contributing to sustainable paddy production. Additionally, the development of a novel hybrid optimization algorithm further enhances the efficiency and effectiveness of disease detection models, ensuring improved performance and robustness in real-world applications.

1.5 Research Hypothesis

The hypothesis of this research is that the implementation of the novel hybrid optimization algorithm, APSWWO, will result in enhanced accuracy for the detection and classification of rice leaf diseases using ANNs. It is expected that by optimizing the parameters of the neural networks through APSWWO, the performance metrics such as accuracy, sensitivity, specificity, precision, and F-measure will significantly improve. This improvement in disease detection and classification will contribute to minimizing yield losses, improving crop health, and ultimately enhancing the productivity and sustainability of rice farming.

1.6 Research Objective

Paddy disease detection using Artificial Intelligence has emerged as a promising approach for enhancing disease management in rice crops. By leveraging the power of AI techniques, such as ANN and computer vision, this research aims to develop an accurate, efficient, and scalable system for early detection and diagnosis of paddy diseases. This section outlines the main objectives of the research, which are:

(i) To develop a model using Artificial Neural Network that can accurately detect paddy diseases at an early stage.

(ii) To develop a novel hybrid optimization algorithm APSWWO to optimize the parameters of ANN efficiently.
(iii) To implement Explainable AI techniques for providing a detailed insight into the decision-making processes of the ANN-based paddy disease detection system.
(iv) To make a comprehensive study among different neural network-based models and optimization algorithms to discover the most effective ones.
(v) To allow farmers and agricultural professionals to take immediate action and implement timely disease management strategies.

1.7 Contribution of this Research

The current work introduces a high-performance paradigm to efficiently address issues in literature surveys related to paddy disease detection. While most studies utilized pre-trained transfer learning models for classifying paddy diseases, this approach may not optimize the models for the specific nuances of paddy disease datasets. To tackle this, the study introduces two customized ANN models: a 26-layer CNN and 13-layer LiteVit models, tailored for efficient paddy disease classification. Additionally, many studies did not use optimization algorithms during ANN training, which can lead to suboptimal performance. To address this, a novel hybrid algorithm, Adaptive Particle Swarm Water Wave Optimization, is proposed to optimize the hyperparameters of the models. Furthermore, some studies worked with relatively small and imbalanced datasets, which can limit model performance. To overcome this, the study collected a comprehensive dataset and applied oversampling techniques for dataset balance. Image enhancement techniques were also employed to optimize image quality. Moreover, the study addressed the limitation of focusing on a limited spectrum of diseases by including ten distinct paddy leaf diseases and a separate class for healthy leaf samples. A comprehensive comparative evaluation involving different configurations of the ANN models was conducted to demonstrate the effectiveness of the proposed hybrid optimization algorithm. Performance metrics such as accuracy, error rate, precision, recall, F1-score, ROC curve, precision-recall curve, and learning curve were considered for comprehensive model evaluation. Overfitting reduction methods were implemented, and model interpretability was enhanced by implementing Grad-CAM in every layer of the models. Overall, the proposed model aims to enable early intervention, effective disease management, and optimized resource utilization to enhance crop productivity and contribute to sustainable paddy production.

1.8 Scope of this Research

This study focuses on classifying ten paddy leaf diseases and a healthy class using a dataset from the Bangladesh Rice Research Institute. Different preprocessing techniques including resizing, normalization, and histogram equalization were applied. A novel hybrid algorithm, Adaptive Particle Swarm Water Wave Optimization (APSWWO), was proposed to optimize hyperparameters of two customized ANN models: a 26-layer CNN and a 13-layer LiteViT. Grad-CAM was employed for interpretability across all layers of both models to gain insights into their decision-making processes and highlight areas of interest in images relevant to paddy disease classification. The research utilized Google Colab for model training and experimentation, leveraging its cloud-based infrastructure for computational resources.

1.9 Thesis Structure

This thesis report is structured as follows:

Chapter 2 Literature Review offers an overview of recent efforts to build models capable of identifying paddy disease and various approaches to the issue.

Chapter 3 Methodology describes the steps and strategy utilized for pre-processing, optimizing, and classifying the models used in the proposed system.

Chapter 4 Results and Analysis presents the results and discussion of this study.

Chapter 5 Conclusion includes the conclusions, limitations of this study, and scopes of future work.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

Extensive research has been conducted to enhance the systems' performance and efficiency in optimizing Artificial Neural Networks and addressing the challenges of paddy disease detection. This section delves into the previous works related to paddy disease detection. This comprehensive analysis of prior research serves as a foundation for understanding the evolution of techniques and methodologies to tackle these complex challenges.

2.2 Previous Works

In recent years, researchers have dedicated significant efforts to leveraging ANNs for paddy disease detection. Numerous studies have delved into various ANN methodologies aimed at enhancing the accuracy and efficiency of disease diagnosis in paddy crops. While many authors have employed ANN techniques, a notable proportion has overlooked the utilization of optimization algorithms. Conversely, a few authors have incorporated single optimization algorithms in their approaches\. This subsection provides an overview of prior studies on ANN-based paddy disease detection, categorizing them into those employing optimization algorithms and those that do not. It elucidates the methodologies and outcomes of research endeavors aimed at enhancing disease detection in paddy crops using Artificial Neural Networks. Table 2.1 and Table 2.2 summarize recent related works of paddy disease detection using ANN.

2.2.1 Paddy Disease Detection using ANN

Numerous authors have employed customized Convolutional Neural Network models for paddy disease detection. Rahman et al. [17] introduced a two-stage compact CNN design and conducted a comparative analysis against state-of-the-art memory-efficient CNN architectures, including MobileNet, NasNet Mobile, and SqueezeNet. The proposed architecture successfully attained the target accuracy of 93.3%. Latif et al. [18] developed a

Deep Convolutional Neural Network (DCNN) approach based on transfer learning for precise detection and classification of paddy leaf diseases. Their modified method utilized a transfer learning technique based on VGG19. The system achieved an impressive average accuracy of 96.08% when using the non-normalized augmented dataset. Additionally, the precision, recall, specificity, and F1-score were found to be 0.9620, 0.9617, 0.9921, and 0.9616, respectively. Tejaswini et al. [19] introduced a 5-layer convolutional model for paddy disease classification and achieved a top accuracy of 78.2%.

Deep learning models are widely employed as feature extraction methods in paddy disease recognition. Sethy et al. [20] employed a dataset containing 5932 paddy leaf images affected by various contaminants. They utilized deep learning architectures, including AlexNet, VGG16, VGG19, GoogleNet, ResNet18, ResNet50, ResNet101, InceptionV3, InceptionResNetV2, DenseNet201, and XceptionNet for extracting deep features from this dataset. The deep features extracted from these models were subsequently classified using Support Vector Machines (SVM). Notably, the combination of deep features from ResNet50 and SVM outperformed the others, achieving an impressive F1 score of 0.9838. Jiang et al. [21] employed Convolutional Neural Networks to extract relevant features from paddy leaf disease images. Subsequently, they utilized the Support Vector Machine method for classification and disease prediction. The SVM model's optimal parameters were determined via a 10-fold cross-validation approach, resulting in an impressive accuracy of 96.8%.

Transfer learning is a machine learning technique where a pre-trained model's knowledge is applied to a different but related problem. It helps leverage existing data and model weights to enhance the performance of new tasks, making it valuable in paddy disease classification. Krishnamoorthy et al. [22] harnessed the power of InceptionResNetV2 alongside a transfer learning strategy for identifying diseases in paddy leaf images. They meticulously optimized the model's parameters for the classification task, achieving a commendable accuracy rate of 95.67%. Chen et al. [23] employed a pre-trained MobileNet-V2 model, trained initially on ImageNet. Their innovative approach resulted in an outstanding average identification accuracy of 99.67% on a publicly available dataset. Matin et al. [24] applied the AlexNet technique to detect the three prevalent paddy leaf diseases: bacterial blight, brown spot, and leaf smut. They got a remarkable outcome (99%) rather than the previous works. Kathiresan et al. [25] proposed a high-accuracy, transfer-learned model that can provide a mobile solution for farmers and agricultural organizations to detect paddy leaf diseases at hand. This study also utilizes a Generative Adversarial Network (GAN) to balance the number of disease samples. The presented model, tested on a GAN-augmented dataset, achieves an average cross-validation accuracy of 98.79%, outperforming paradigm classification architectures. Hasan et al. [26] employed four models, namely VGG-19, Inception-Resnet-V2, ResNet-101, and Xception, and achieved the highest accuracy with Inception-ResNet-V2, reaching 92.68%. Bari et al. [27] developed a Faster Region-Based Convolutional Neural Network (Faster R-CNN) for real-time paddy leaf disease detection. Their approach demonstrated high effectiveness in diagnosing three distinct paddy leaf diseases: paddy blast, brown spot, and hispa, achieving accuracies of 98.09%, 98.85%, and 99.17%, respectively.

Table 2.1: Recent related works of paddy disease detection using ANN (without optimization)

Paper	Dataset	Description of Methods	Result	Similarity with this Research	Dissimilarity with this Research	Limitation
Rahman et al. [17]	Real life data False Smut 93 Brown Planthopper (BPH) 71 BLB 138 Neck Blast 286 Stemborer 201 Hispa 73 Sheath Blight and/or Sheath Rot 219 Brown Spot 111 Others 234	Classifier: Two stage CNN model. Model interpretability techniques used in first and last layers of CNN.	Accuracy 93.3%	CNN technique. Model interpretability technique.	Optimization algorithm. Image Enhancement Techniques. Model interpretability techniques in all layers.	Do not use any image enhancement technique. Do not use any optimization algorithm. Focus on only one performance metric. Model interpretability techniques focus on only two layers. Dataset is small. Imbalance dataset.
Sethy et al. [20]	Real life data Bacterial blight 1584 Paddy Blast 1400 Brown spot 1600 Tungro 1308	Feature Extraction Method: AlexNet VGG16 VGG19, GoogleNet ResNet18 ResNet50 ResNet101 InceptionV3 InceptionResNetV2 DenseNet201 Classifier: SVM	Accuracy 98.38% Specificity 98.46% Recall 98.38% F1-score 98.38%	ANN technique	Optimization algorithm. Model interpretability technique. Overfitting reducing method. Oversampling: SMOTE Classifier: CNN	Imbalance dataset. Do not use model interpretability techniques. Do not use any overfitting reducing method. Possibility of having premature convergence. Do not use any optimization algorithm. Focus on only four diseases. Overlooked the healthy class.
Jiang et al. [21]	Real life data Healthy 2274 Paddy blast 1634 BLB 1765 Streak leaf spot 1678 Sheath blight 1560	Feature Extraction Method: CNN Classifier: SVM Overfitting Reducing Methode: 10 fold cross validation	Accuracy 96.8%	CNN Overfititting Reducing Method	Optimization algorithm. Model interpretability technique. Oversampling: SMOTE Classifier: CNN Overfititting Reducing Method: Early stopping Dropout	Do not use any optimization algorithm. Focus on only one performance metrics. Imbalance dataset. Do not use any image enhancement algorithm. Focus on only five diseases.
Krishnamoorthy et al. [22]	Kaggle dataset BLB 1300 Brown spot 1300 Blast 1300 Healthy 1300	Classifier: InceptionResNetV2 Model Interpretability techniques of hidden layers	Accuracy 95.67%	ANN	Classifier: CNN Model Interpretability techniques of all layers. Optimization algorithm. Image Enhancement Techniques.	Do not use any image enhancement technique. Do not use any optimization algorithm. Focus on only one performance metric. Model interpretability techniques focus on only hidden layers. . Focus on only four diseases.

Paper	Dataset	Description of Methods	Result	Similarity with this Research	Dissimilarity with this Research	Limitation
Chen et al. et al. [23]	Online & real life dataset Stackburn 42 Leaf smut 61 Leaf scald 57 False smut 46 Blast 219 Stem rot 50 White tip 32 Sheath rot 72 Stripe blight 48 Sheath blight 86 Bacterial leaf streak 192 Brown spot 202	Classifier: MobileNet-V2	Accuracy 99.67%	ANN	Optimization algorithm. Model interpretability technique. Overfitting reducing method. Oversampling: SMOTE Classifier: CNN	Imbalance dataset. Focus on only one performance metric. Do not use model interpretability techniques. Do not use any overfitting reducing method. Possibility of having premature convergence. Do not use any optimization algorithm. Overlooked healthy class.
Martin et al. [24]	Kaggle dataset Bacterial blight 40 images Brown spot 40 images Leaf smut 40 images	AlexNet	Accuracy 99%	ANN	Classifier: CNN Optimization algorithm. Overfitting reducing method. Model interpretability techniques.	Small dataset. Do not use model interpretability techniques. Do not use any overfitting reducing method. Focus on only one performance metric. Do not optimization algorithm. Overlooked healthy class. Focus on only three diseases.
Kathiresan et al. [25]	Online dataset Bacterial leaf streak 776 False smut 1056	GAN	Accuracy 98.38%	ANN technique	Classifier: CNN Optimization algorithm. Overfitting reducing method. Model interpretability techniques.	Small dataset. Only focus on two classes. Overlooked healthy class. Do not use model interpretability techniques. Do not use any overfitting reducing method. Do not use any optimization algorithm.
Hasan et al. [26]	UCI and Kaggle dataset Brown spot 166 Leaf blast 159 Leaf blight 216 Leaf smut 216 Healthy 227	VGG-19 Inception-Resnet-V2 ResNet-101 Xception	Accuracy 92.86% Precision 92.15% Recall 90.56% F1-score 90.56%	ANN technique	Classifier: CNN Optimization algorithm. Overfitting reducing method. Model interpretability techniques.	Small dataset. Imbalance dataset Only focus on five classes. Do not use model interpretability techniques. Do not use any overfitting reducing method. Do not use any optimization algorithm.

Paper	Dataset	Description of Methods	Result	Similarity with this Research	Dissimilarity with this Research	Limitation
Bari et al. [27]	Real life data Blast 500 Brown spot 500 Hispa 500 Healthy 500	Faster R-CNN	Accuracy 99.17%	ANN techniques	Classifier: CNN Optimization algorithm. Overfitting reducing method. Model interpretability techniques.	Small dataset. Only focus on five classes. Do not use model interpretability techniques. Do not use any overfitting reducing method. Do not use any optimization algorithm.
Latif et al. [18]	Healthy 371 Narrow brown spot 352 leaf scald 358 Leaf blast 363 Brown spot 373 Bacterial leaf blight 350	Classifier: CNN	Accuracy 96.08% Precision 0.9620 Recall 0.9617 Specificity 0.9921 F1-score 0.9616	ANN techniques	Classifier: CNN Optimization algorithm. Overfitting reducing method. Model interpretability techniques.	Small dataset. Imbalance dataset Only focus on six classes. Do not use model interpretability techniques. Do not use any overfitting reducing method. Do not use any optimization algorithm. Do not use any image enhancement techniques.
Tejaswini et al. [19]	Brown spot 400 Hispa 500 Blast 300 Healthy 400	Classifier: CNN	Accuracy 78.2%	ANN techniques	Classifier: CNN Optimization algorithm. Overfitting reducing method. Model interpretability techniques.	Small dataset. Imbalance dataset Only focus on four diseases. Do not use model interpretability techniques. Do not use any overfitting reducing method. Do not use any optimization algorithm. Do not use any image enhancement techniques.

Various researchers have explored the use of transfer learning with CNNs for paddy disease detection, including Rahman et al. [17], Sethy et al. [20 and Tejaswini et al. [19]. However, some studies [17-19] have utilized small and imbalanced datasets. Notably, image enhancement techniques were not incorporated by Latif et al. [18] and Tejaswini et al. [19], which presents a notable research gap affecting model performance. Additionally, model interpretability, crucial for enhancing trust and transparency, has been overlooked by several studies [18-20]. Furthermore, disparities exist in the optimization of ANN model parameters, with some studies neglecting optimization algorithms altogether, while others employ single optimization techniques. This highlights the need for comprehensive parameter optimization strategies in this research domain.

2.2.2 Paddy Disease Detection using ANN with Optimization

Some researchers have employed a single optimization algorithm in paddy disease detection. Wang et al. [28] introduced an attention-based depthwise separable neural network combined with Bayesian optimization (ADSNN-BO) for the detection and classification of paddy diseases in paddy leaf images. Their method achieved an impressive accuracy of 94.65%. Ramesh et al. [29] developed an optimized deep neural network using the Jaya algorithm to recognize and classify paddy leaf diseases. Their study directly captured images of paddy plant leaves from the farm field, covering conditions like bacterial blight, brown spots, sheath rot, and blast diseases. Their model achieved impressive accuracy rates: 98.9% for blast-affected leaves, 95.78% for bacterial blight, 92% for sheath rot, 94% for brown spots, and 90.57% for normal leaf images. Sharma et al. [16] introduced a method for the automated evolution of CNN architecture by employing the Big Bang–Big Crunch (BB–BC) algorithm. This CNN architecture was tested with 5932 real-world images of paddy leaves affected by bacterial leaf blight, paddy blast, brown leaf spot, and tungro diseases, resulting in an impressive accuracy of 98.7%. A novel approach for early detection and identification of paddy leaf diseases in the Thanjavur region utilizes the Radial Basis Function Neural Network (RBFNN) classifier, optimized with the Salp Swarm Algorithm (SSA) [30]. This research addresses four categories of infected paddy images (bacterial blast, bacterial blight, leaf tungro, and brown spot) alongside a set of normal images. Additionally, Raja et al. [31] proposed an AI-assisted disease detection method using RBFNN with the Salp-Swarm Algorithm (PLDC-RBFNN-SSA) for paddy leaf disease segmentation. The PLDC-RBFNN-SSA approach demonstrates significantly improved accuracy, with increases of 15.25%, 18.98%, 20.5%, and 24.85% across various disease categories.

Table 2.2: Recent related works of paddy disease detection using ANN (with optimization)

Paper	Dataset	Description of Methods	Result	Similarity with this Research	Dissimilarity with this Research	Limitation
Sharma et al. [16]	Real life data Bacterial Leaf Blight (BLB) 1584 Paddy Blast 1400 Brown spot 1600 Tungro 1308	Classifier: CNN Optimization Algorithm: BB-BC	Accuracy 98.70% Precision 98.81% Recall 98.75% F1-score 98.70%	ANN technique	Hybrid optimization algorithm. Model interpretability technique. Overfitting reducing method. Oversampling: SMOTE	Imbalance dataset. Do not use model interpretability techniques. Focus on only four diseases. Overlooked the healthy class. Do not use any overfitting reducing method. Possibility of having premature convergence.
Raja et al. [31]	Bacterial Blast Bacterial Blight Leaf Tungro Brown Spot	Classifier: RBFNN Optimization Algorithm: SSA	Time is reduced by 50.2%, 48.2%, 38.26%, 20.2%	Optimization algorithm	Optimizer: APSWWO Classifier: CNN Model interpretability technique.	Do not use model interpretability techniques. Do not use any overfitting reducing method. Focus on only four diseases.

Paper	Dataset	Description of Methods	Result	Similarity with this Research	Dissimilarity with this Research	Limitation
Wang et al. [28]	Online data Brown spot 523 Hispa 565 Blast 779 Healthy 503	Classifier: ADSNN Optimization Algorithm: BO Model interpretability technique of some laayers	Accuracy 94.65% Precision 92.6% Recall 87.4% F1-Score 89.6%	ANN	Overfitting reducing method. Oversampling: SMOTE Classifier: CNN Optimization algorithm: APSWWO	Small dataset. Imbalance dataset. Do not use model interpretability techniques in all layers. Do not use any overfitting reducing method. Possibility of having premature convergence. Do not use any image enhancement algorithm. Focus on only four diseases.
Ramesh et al. [29]	Real life dataset Healthy 95 BLB 125 Blast 170 Sheath rot 110 Brown spot 150	Classifier: DNN Optimization Algorithm: Jaya	Accuracy 98.9%	Optimization algorithm CNN	Optimizer: APSWWO Classifier: CNN Overfirring reducing method. Model interpretability technique.	Small dataset. Imbalance dataset. Focus on only one performance metric. Do not use model interpretability techniques. Do not use any overfitting reducing method. Possibility of having premature convergence. Do not use any image enhancement algorithm. Focus on only five diseases. Overlooked healthy class.
Devi et al. [30]	Bacterial Blast Bacterial Blight Leaf Tungro Brown Spot	Classifier: RBFNN SVM Optimization Algorithm: SSA	Accuracy 98.47%	Optimization algorithm	Optimizer: APSWWO Classifier: CNN Overfirring reducing method. Model interpretability technique.	Do not use model interpretability techniques. Do not use any overfitting reducing method. Possibility of having premature convergence. Focus on only four diseases. Overlooked healthy class.

While Rahman et al. [17], Jiang et al. [21], Krishnamoorthy et al. [22], Chen et al. et al. [23], Martin et al. [24], and Bari et al. [23] have emphasized a single performance metric, specifically accuracy, it's important to note that focusing solely on accuracy may not provide a comprehensive evaluation of model performance. A holistic assessment of model performance could include additional metrics for other aspects of model behavior. Moreover, even when achieving high accuracy, Sethy et al. [20], Jiang et al. [21], and Wang et al. [28] did not employ any overfitting reduction methods. Overfitting is a critical concern in ML, and addressing it ensures the generalizability of models to unseen data. Additionally, Sharma et al. [16], Sethy et al. [20], Jiang et al. [21], Krishnamoorthy et al. [22], Wang et al. [28], Ramesh et al. [29], Martin et al. [24], Kathiresan et al. [25], Hasan et al. [26], Bari et al. [27], Latif et al. [18], and Tejaswini et al. [19] have primarily concentrated on the classification of 2-7 classes of paddy diseases. To enhance the practical utility of disease detection models, expanding the scope to consider a broader range of disease classes commonly encountered in real-world scenarios may be beneficial.

2.3 Summary

The study introduces customized ANN models (26-layer CNN and 13-layer LiteVit) to enhance paddy disease classification, addressing limitations of pre-trained transfer learning models. Highlighting the importance of optimization algorithms, it proposes a hybrid APSWWO algorithm for parameter optimization, crucial for model efficiency and accuracy. Dataset imbalance and size constraints are mitigated through oversampling and preprocessing techniques, ensuring model robustness. Unlike previous works, the study emphasizes comprehensive evaluation metrics and model interpretability, addressing gaps in existing research. By bridging these limitations, the proposed model offers early disease intervention, resource optimization, and sustainable paddy production enhancement.

CHAPTER 3

METHODOLOGY

3.1 Introduction

This research aims to create optimized Artificial Neural Network models for predicting paddy disease by introducing a novel hybrid optimization algorithm. The study unfolds through seven key phases: image acquisition, image preprocessing, over-sampling to balance the dataset, the development of nature-inspired hybrid optimization methods, the utilization of Artificial Neural Network, incorporating model interpretability techniques, and a comprehensive performance assessment for paddy disease classification. This section delves into an in-depth discussion of the proposed novel hybrid optimization algorithm and its role in optimizing ANN hyperparameters for paddy disease prediction, providing a step-by-step overview of the entire research process.

3.2 Proposed Model of Paddy Disease Detection

The process of analyzing paddy disease is briefly represented in Figure 3.1. It commences with compiling details from paddy disease images, followed by preprocessing and balancing these images. The proposed hybrid model is then leveraged to fine-tune the ANN parameters using the disease images. After assessing the performance through training, testing, and validation and comparing various forms of ANN algorithms (ANN_Original, ANN_PSO, ANN_APSO, ANN_WWO, and ANN_APSWWO), the final model is selected for disease prediction. The workflow diagram of paddy disease recognition system is briefly represented in Figure 3.1

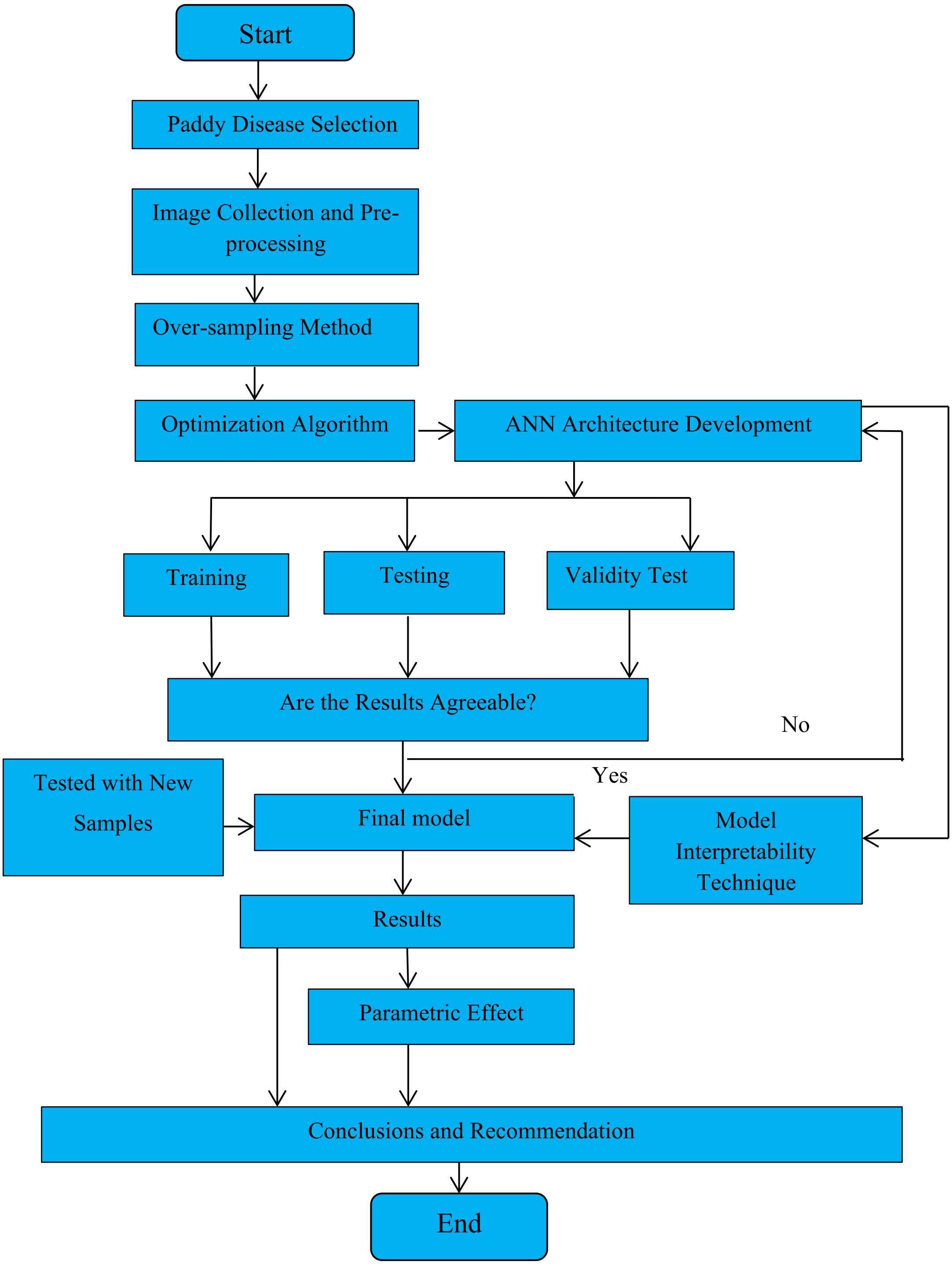


Figure 3.1: Workflow diagram of paddy disease recognition system

3.2.1 Data Collection

Paddy diseases are a group of diseases that affect the leaves of paddy plants, leading to significant yield losses and reduced crop quality. Various pathogens, including fungi, bacteria, and viruses, cause these diseases. Different types of paddy leaf disease are Bacterial Leaf Blight (BLB), Brown Planthopper (BPH), brown spot, false smut, hispa, leaf scaled, neck blast, tungro, sheath blight rot, stemborer, blast, black grain, kernel smut, and Green Leafhopper (GLH). This study has selected ten paddy leaf diseases: BLB, BPH, brown spot, false smut, hispa, leaf scaled, neck blast, tungro, sheath blight rot, and stemborer and one healthy class to classify disease from healthy leaf samples efficiently. The dataset utilized in this study comprises a total of 2,483 images collected from the Bangladesh Rice Research Institute (BRRI). The specific distribution of images within each class is detailed in Table 3.1. Figure 3.2 illustrates representative samples from the dataset.

Table 3.1: Dataset description

Paddy Disease	**Description**		**No. of Collected Data**
	Pathogens	**Symptoms**	
BLB	Xanthomonas oryzae pv. oryzae (bacterium)	Water-soaked lesions, gray-green streaks, leaf wilting, yellowing, drying leaves	138
BPH	Nilaparvata lugens (insect)	Stunted growth, yellowed leaves, "Hopper burn" damage, honeydew secretion	71
Brown Spot	Cochliobolus miyabeanus (fungus)	Small, dark spots, enlarged centers, yellow halos, reduced yield	201
False Smut	Ustilaginoidea virens (fungus)	False smut balls, gray-green spore masses, infected grain appearance, reduced grain quality	93
Hispa	Dicladispa armigera (insect)	Shot-hole damage, skeletonized leaves, stunted growth, yellowing leaves	73
Leaf Scaled	Coccidae family of scales and mealybugs (insects)	Sooty mold, honeydew secretion, reduced vigor, leaf deformation	217
Neck Blast	Magnaporthe oryzae (fungus)	Lesions on neck area, blast infection, yield reduction, empty grains	558
Paddy Tungro	Paddy tungro bacilliform virus (RTBV) and Paddy tungro spherical virus (RTSV)	Stunting, yellowing leaves, reduced tillering, mottling of leaves	195
Sheath Blight Rot	Rhizoctonia solani (fungus)	Sheath lesions, rotting sheaths, yield reduction, white mycelium	502
Stemborer	Scirpophaga and Chilo (insects)	Tunneling in stems, wilting plants, dead hearts, reduced yield	201
Healthy Plant	No pathogen	No symptoms	234

(a) Bacterial Leaf Blight (b) Brown Planthopper (c) Brown Spot (d) False Smut

(e) Hispa (f) Leaf Scaled (g) Neck Blast (h) Tungro

(i) Sheath Blight Rot (j) Stemborer (k) Healthy Plant

Figure 3.2: Dataset samples

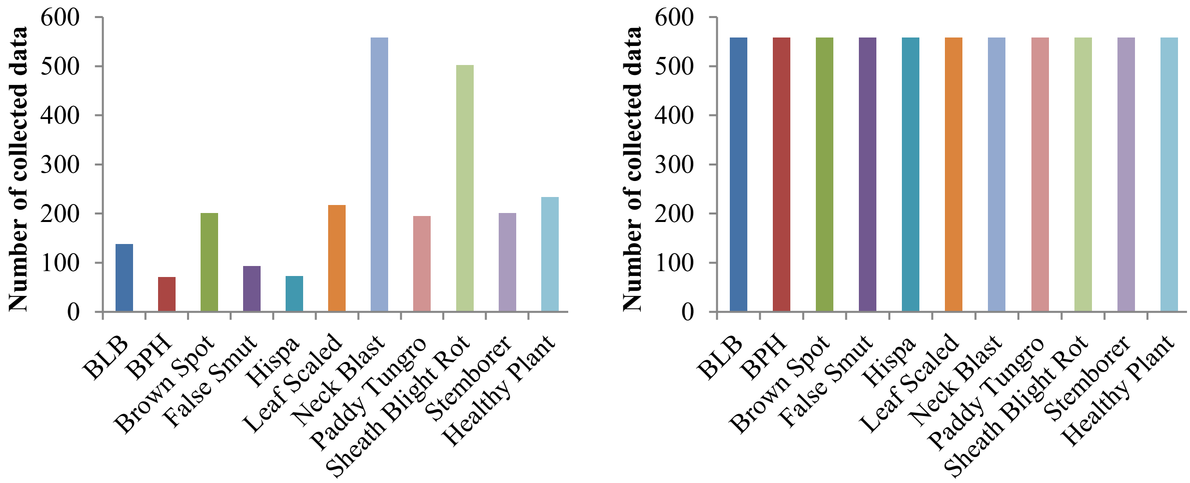
3.2.2 Data Pre-processing

Due to differing qualities within the image dataset, they have been resized (128×128) and subjected to min-max normalization to guarantee consistency. Moreover, Adaptive Histogram Equalization has improved the local contrast in the images. Figure 3.3 shows the image dataset's pre-processing steps, including resizing, normalization, and Adaptive Histogram Equalization. This study applied the Synthetic Minority Over-sampling Technique to address the class imbalance within the dataset. SMOTE is utilized in various domains, such as fraud detection, medical diagnosis, image classification, and text sentiment analysis. This study implements SMOTE to equalize sample numbers across all classes. Initially, the "neck blast" class had the highest samples (558). With SMOTE, the dataset was balanced, ensuring each of the 11 classes had 558 images. The procedure involved selecting a sample from the minority class, identifying its k-nearest neighbors, and creating synthetic examples through linear interpolation. This process resulted in balanced class distribution, enhancing representation during training. Mathematically, SMOTE utilizes weighted feature value averages for computing synthetic examples using Eq.(3.1):

$$\text{New Example} = \text{Example} + (\text{Random Value}) * (\text{Neighbor - Example}) \quad (3.1)$$

Consequently, the "neck blast" class, initially comprising 558 images, was aligned with the other classes, which now contained 558 images. This balanced dataset facilitates more robust and unbiased machine learning analyses.

Figure 3.3: Image resizing, Normalization, and Adaptive Histogram Equalization



(a) Before SMOTE (b) After SMOTE

Figure 3.4: Distribution of number of collected data

3.2.3 Artificial Neural Network

An Artificial Neural Network is a computer structure that studies how biological neurons in the brain process information. The neural network's strength comes from its capacity to learn from data and extrapolate from it to predict future values of the same kind. The system derives an output from a generalization of structured input data. This study focused on the 26-layer Convolutional Neural Network and 13-layer Lightweight Vision Transformer.

a) Convolutional Neural Network

Convolutional Neural Networks are deep learning models specifically designed for image and spatial data analysis. They excel in various applications, including image classification, object detection, facial recognition, and medical image analysis, because they can automatically learn and extract hierarchical features from data [37, 38]. This study employs a 26-layer CNN model to classify paddy diseases efficiently. Each layer is meticulously designed to capture and extract valuable features from the input data, enabling precise classification. The architecture of the proposed 26-layer CNN model is depicted in Figure 3.5.

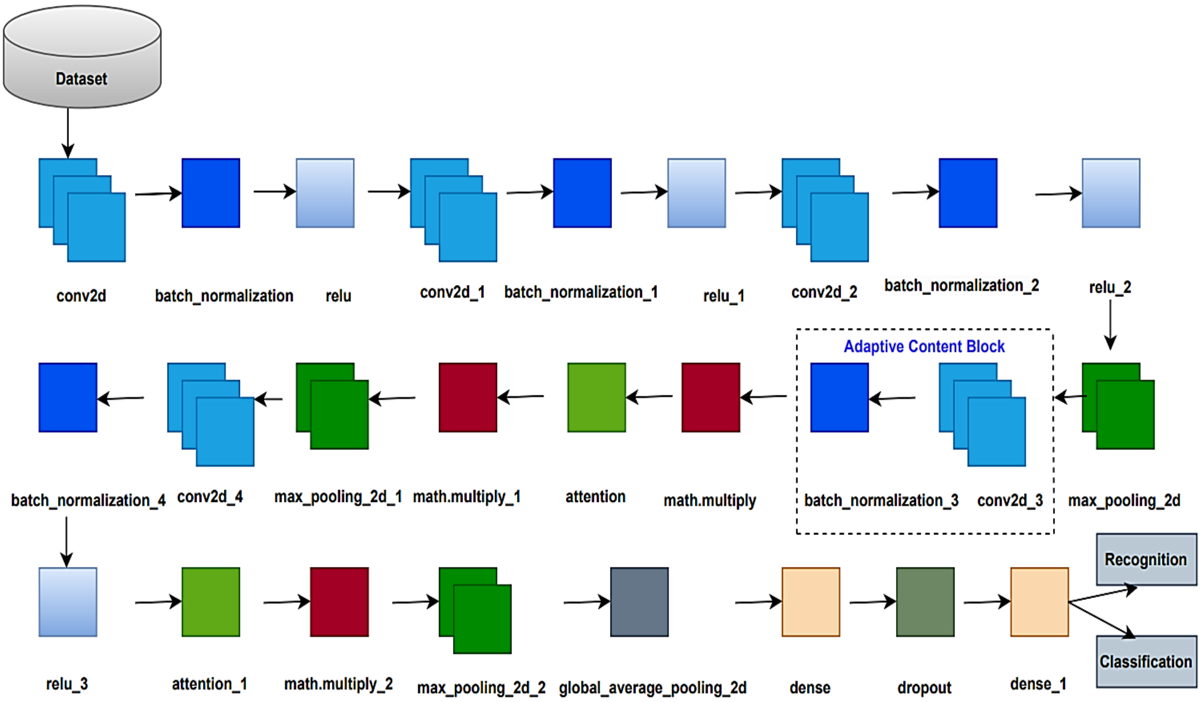


Figure 3.5: Architecture of proposed 26-layer CNN model

The model comprises four Conv2D layers, each followed by BatchNormalization and Rectified Linear Unit (ReLU) activation to enhance feature extraction. The Conv2D layers are responsible for feature extraction through convolution operations. The ReLU layers introduce non-linearity by applying the Rectified Linear Unit activation function. The BatchNormalization layers help to stabilize training and accelerate convergence by normalizing the inputs to each layer. The model also features three MaxPooling2D layers, which progressively downsample the spatial dimensions of the data. The model incorporates a unique component known as the "Adaptive Context Block," which includes a Conv2D layer and a BatchNormalization layer with an activation function to modify the input features adaptively. The network also integrates two Attention layers, which utilize self-attention mechanisms to weigh the importance of different features. These attention mechanisms improve the model's ability to recognize complex patterns and relationships within the data. The three math.multiply layers play a crucial role in modulating feature representations within the neural network by performing element-wise multiplications and enhancing the network's ability to capture important spatial relationships and dependencies in the data. The Global Average Pooling layer aggregates features from the Convolutional and Attention layers. The fully connected layers (Dense) refine the features before the final classification. Dropout layers prevent overfitting, and the output layer employs softmax activation for multi-class classification. The overall summary of the proposed 26-layer CNN model is included in Table 3.2.

Table 3.2: Proposed 26-layer CNN model summary

CNN Layers	Output Shape	Parameters
conv2d	(None, 128, 128, 64)	1792
batch_normalization	(None, 128, 128, 64)	256
relu	(None, 128, 128, 64)	0
con2d_1	(None, 128, 128, 128)	73856
batch_normalization_1	(None, 128, 128, 128)	512
relu_1	(None, 128, 128, 128)	0
conv2d_2	(None, 128, 128, 256)	295168
batch_normalization_2	(None, 128, 128, 256)	1024
relu_2	(None, 128, 128, 256)	0
max_pooling2d	(None, 64, 64, 256)	0
conv2d_3	(None, 64, 64, 256)	65792
batch_normalization_3	(None, 64, 64, 256)	1024
math.multifly	(None, 64, 64, 256)	0
attention	(None, 64, 64, 256)	1
math.multifly_1	(None, 64, 64, 256)	0
max_pooling2d_1	(None, 32, 32, 256)	0
conv2d_4	(None, 32, 32, 256)	590080
batch_normalization_4	(None, 32, 32, 256)	1024
relu_3	(None, 32, 32, 256)	0
attention_1	(None, 32, 32, 256)	0
math.multifly_2	(None, 32, 32, 256)	0
max_pooling2d_2	(None, 16, 16, 256)	0
global_average_pooling2d	(None, 256)	0
dense	(None, 128)	32896
dropout	(None, 128)	0
dense_1	(None, 11)	1419

Trainable Parameters	**1062924**
Non-trainable Parameters	**1920**
Total Parameters	**1064844**

b) Lightweight Vision Transformer

Lightweight Vision Transformer architecture designed for efficient image classification tasks. Its innovative approach combines the strengths of CNNs and transformer models, making it suitable for various applications, such as real-time image recognition on edge devices, object detection, and natural language processing [39]. This study leverages a LiteViT model composed of thirteen distinct layers, each contributing to the model's overall architecture and functionality. The architecture of proposed 13-layer LiteViT model is depicted in Figure 3.6. Among these layers, the model includes three conv2d layers responsible for conducting convolutional operations. These layers play a pivotal role in feature extraction and pattern recognition, allowing the network to capture meaningful information from the input data. Furthermore, the LiteViT model incorporates two max_pooling2d layers, which are essential for downsampling the spatial dimensions of the feature maps, aiding in the reduction of computational complexity while preserving valuable features. These layers contribute to the model's ability to process information efficiently. In addition to these layers, the LiteViT model employs attention mechanisms represented by the layers "attention" and "attention_1." Attention mechanisms enhance the model's capacity for focusing on relevant information and are particularly effective in various tasks such as image classification and natural language processing. Operators like "operators.add" and "operators.add_1" are integral to the LiteViT model, enabling the aggregation and combination of features extracted from different layers. This facilitates the model's ability to consolidate and refine information, improving its overall performance. The "global_average_pooling2d" layer plays a role in spatial feature reduction, promoting efficient information processing by computing the average values of the feature maps. Following this, the LiteViT model features "dense" layers responsible for dense (fully connected) connections, and "dropout" layers that mitigate the risk of overfitting. The "dense_1" layer, which marks the final layer of the LiteViT model, is essential for generating output predictions, making it a critical component for the successful completion of paddy disease classification tasks. The overall summary of proposed 13-layer LiteViT model is included in Table 3.3.

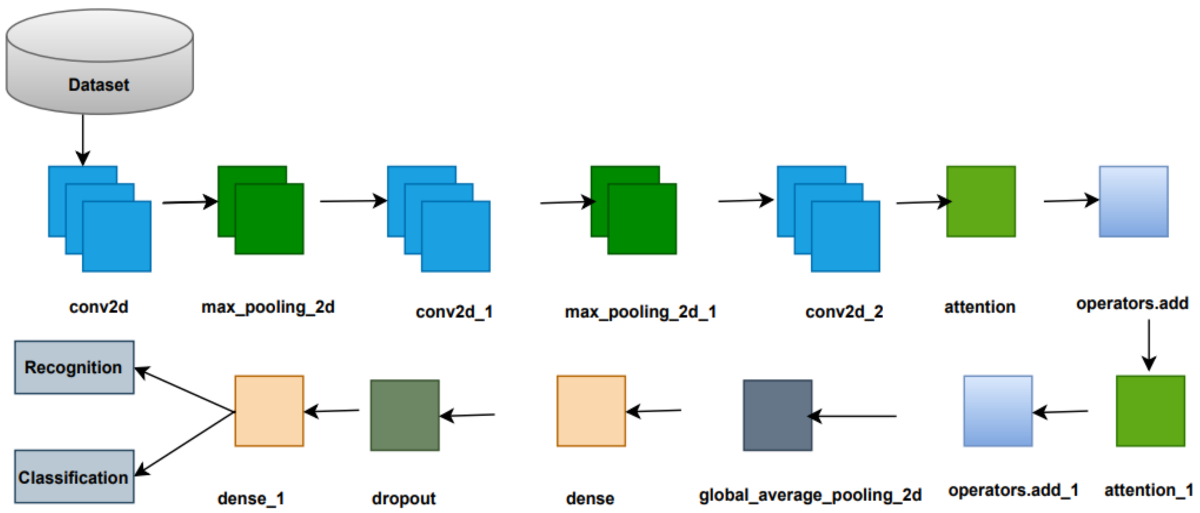


Figure 3.6: Architecture of proposed 13-layer LiteViT model

Table 3.3: Proposed 13-layer LiteViT model summary

LiteViT Layers	Output Shape	Parameters
conv2d	(None, 128, 128, 64)	1792
max_pooling2d	(None, 64, 64, 64)	0
conv2d_1	(None, 32, 32, 64)	73856
max_pooling2d_1	(None, 16, 16, 64)	0
conv2d_2	(None, 8, 8, 64)	295168
attention	(None, 8, 8, 64)	263168
operators.add	(None, 8, 8, 64)	0
attention_1	(None, 8, 8, 64)	263168
operators.add_1	(None, 8, 8, 64)	0
global_average_pooling2d	(None, 256)	0
dense	(None, 512)	131584
dropout	(None, 512)	0
dense_1	(None, 11)	2827
Trainable Parameters		**768395**
Non-trainable Parameters		**350**
Total Parameters		**768745**

3.2.4 Nature-inspired Optimization Algorithms

Nature-inspired algorithms are optimization methods that handle challenging optimization issues by emulating the behavior of natural systems. These algorithms effectively locate optimal solutions for multi-dimensional and multi-modal problems. This research proposed a novel hybrid optimization algorithm, combining two nature-inspired optimization algorithms: Adaptive Particle Swarm Optimization and Water Wave Optimization.

a) Particle Swarm Optimization Algorithm

PSO is a heuristic algorithm that attempts to optimize a given problem by gradually improving a candidate solution to a given metric of that particular solution's quality, also known as the fitness function. It is based on the social foraging behavior of bird flocks and fish schools. Each member of the PSO's population, known as the swarm, is addressed as a particle, representing a potential solution in the multi-dimensional search space. Particles initially populate different locations at random and migrate in the direction of the ideal outcome. Every particle maintains its p_{best}, or best possible position, in memory. Similarly to that, g_{best} refers to the swarm's ideal position. The particle uses both of these in determining its next best position in the swarm. Let's say a swarm consists of N particles as (p_1, p_2, p_3, …, p_n). The position of the particle in the d-dimensional space is denoted as $X_p = (X_{p1}, X_{p2}, X_{p3}, …, X_{pd})$. A velocity vector is associated with every particle moving, denoted as $V_p = (V_{p1}, V_{p2}, V_{p3}, …, V_{pd})$ at different positions in d-dimensional space. The velocity and position of the p-particle can then be updated using the Eq. (3.2) and Eq. (3.3).

$$v_p = w_i v_p + c_1 * r_1(p_{bestp} - X_p) + c_2 * r_2(g_{best} - X_p) \tag{3.2}$$

$$X_p = X_p + V_p \tag{3.3}$$

Where, r_1 and r_2 are evenly distributed random values in the range (0, 1) and c_1 and c_2 are positive learning constants known as the cognitive and social scaling parameters. The inertia weight w_i, regulates how the past affects the present velocity. Up until an ideal solution is found, the locations and speeds of each particle are computed for each iteration. The algorithm of PSO is depicted in Algorithm 3.1.

Algorithm 3.1: Algorithm of PSO	
1.	For each particle, initialize population
2.	For each particle, do
3.	Calculate fitness value
4.	If the fitness value is better than the best fitness value (p_{best}) in history
5.	Set the current value as the new p_{best}
6.	Choose the particle with the best fitness value of all the particles as the g_{best}
7.	Calculate particle velocity according to Eq. (3.2)
8.	Update particle position according to Eq. (3.3)
9.	Terminate While maximum iterations or minimum error criteria are attained

b) Adaptive Particle Swarm Optimization Algorithm

Numerous PSO-based test data generation methodologies have been suggested. PSO offers rapid convergence but struggles with a low exploitation rate and local minima trapping. While various PSO versions aim to improve this, simultaneously achieving both objectives remains challenging. PSO heavily relies on inertia weight to balance local and global search. A high inertia weight promotes global exploration, while a low value enhances local exploitation. Adaptive inertia weight techniques aim to counteract this issue by adjusting inertia weight based on particle fitness. The ratio α, is calculated as the particle's fitness over the swarm's average fitness, adapts the inertia weight. Eq. (3.4) and Eq. (3.5) detail this adjustment using particle fitness.

$$f_{avg} = \sum \frac{f_j}{n} \tag{3.4}$$

$$\alpha_j = \frac{f_j}{f_{avg}} \tag{3.5}$$

Where, f_j = fitness of j^{th} particle and n is the number of particles in the swarm. The value of α_j is more than 1 for individuals whose current fitness is more than the average fitness of the swarm. These particles' performance is better and closer to the optimal solution. So, a smaller inertia weight should be used and calculated by Eq. (3.6).

$$w_i = w_{min} + (w_{max} - w_{min}) * (f_j - f_{avg})/(f_{max} - f_{avg}) \tag{3.6}$$

The value of α_j is less than 1 when the particle's fitness is less than the average fitness of the swarm. Increasing inertia weight can strengthen the particle's search capability and enables it to jump out of the local minimum. So, a higher inertia weight should be used and calculated by Eq. (3.7).

$$w_i = w_{min} + (w_{max} - w_{min}) * (1 + X)/e^{\alpha} \tag{3.7}$$

where X is a constant. The experimental results show that at X = 1.4 particle convergence rates are better. The algorithm of APSO is depicted in Algorithm 3.2.

Algorithm 3.2: Algorithm of APSO

1. Initialize the swarm of particles for PSO.
2. Initialize particle Velocity $v_{p,}$ and position X_p
3. Initialize p_{best}, and g_{best} for each particle
4. Set p_{best} for each particle initially as its current position.
5. Set g_{best}as as the best position among all the particles in the swarm.
6. Repeat until a termination condition is met:
7. Evaluate the fitness of f_j for each particle in the APSO population using the fitness function.
8. Average fitness is to be calculated for APSO using Eq. (3.4)
9. The ratio αj is calculated using Eq. (3.5)
10. If $\alpha_j \geq 1$ $then$
11. Calculate w_i using Eq. (3.6)
12. Else calculate w_i using Eq. (3.7)

13. Update the velocity and position of the particles using Eq. (3.2) and (3.3)
14. Find the value of $currp_{best}, currg_{best}$ according to $New\ v_p$ and $New\ X_p$
15. For each particle in the population, do
    If $currp_{best} > p_{best}$ then
    $$p_{best} = currp_{best}$$
16. If $currg_{best} > g_{best}$ then
    $$g_{best} = currg_{best}$$
17. If each particle has been processed, select the best individuals.

---

c) Water Wave Optimization

Shallow water wave models guide the WWO for addressing optimization problems. Let's consider a maximization problem with an objective function f. In the WWO, the seabed area corresponds to the solution space Y, and a point's fitness depends on its depth relative to the still water level. This fitness is denoted as f(y), and proximity to the water level signifies better fitness. The WWO maintains a population of solutions, akin to "waves" with a given height (h) and wavelength (λ). Initially, h is set to a constant value (h_{max}), and each wave has a wavelength of 0.5. Throughout problem-solving, three types of wave operations are considered: Propagation, Refraction, and Breaking. The algorithm of WWO is depicted in Algorithm 3.3. Each wave must travel exactly once in each generation. By moving each dimension d of the initial wave y as the propagation operator produces a new wave y_0 as Eq.(3.8).

$$y'(d) = y(d) + \text{rand}(-1,1).\ \Lambda\ L(d) \tag{3.8}$$

where rand(-1,1) is a uniformly distributed random number within the range (-1,1), and L(d) is the length of the dth dimension of the search space. If a new position falls outside the viable range, it will be randomly reset to a position within that range. As a wave transitions from deep water (characterized by poor fitness) to shallow water (with high fitness), its wavelength decreases, and its wave height increases. Consequently, we evaluate the fitness of the daughter wave, denoted as y_0, after the propagation step. In the population, the wave y is replaced by y_0 if f(y) is less than $f(y_0)$, and the wave height of y_0 is reset to h_{max}. In all other scenarios, y remains unchanged, but its height h decreases by one, emulating the loss of energy due to inertial resistance, vortex shedding, and bottom friction. The wavelength of each wave y is updated after each generation according to Eq.(3.9).

$$\lambda = \lambda.\alpha^{(f(x)-f(\min)+\varepsilon)/\,(f(max)-f(\min)+\varepsilon)} \tag{3.9}$$

where f_{max} and f_{min} are, respectively the maximum and minimum fitness values among the current population, α is the wavelength reduction coefficient, and ϵ is a minimal positive number to avoid division-by-zero. Eq. (3.9) ensures that higher fitness waves have smaller wavelengths and thus propagate within smaller ranges.

During wave propagation in WWO, if a wave ray is not perpendicular to the isobath (a line connecting points with the same water depth), its direction changes. Wave rays are observed to diverge in deep areas and converge in shallow regions. In the WWO algorithm, refraction is applied only to waves whose heights have dropped to zero. The position following refraction is determined using a Eq.(3.10).

$$y'(d)= \mathrm{N}\left(\frac{y_*(d)+y(d)}{2}, \frac{|y_*(d)+y(d)|}{2}\right) \tag{3.10}$$

where, y* is the best solution found so far, and N(μ,σ) is a Gaussian random number with mean μ and standard deviation σ. After refraction, the new position is calculated as a random number, which is centered halfway between the original position and the known best position. The standard deviation is equal to the absolute value of the difference between these two positions. Additionally, the wave height of y_0 is reset to h_{max}, and its wavelength is set accordingly Eq. (3.11).

$$\lambda' = \lambda \frac{f(y)}{f'(y)} \tag{3.11}$$

When a wave reaches a location where the water depth falls below a specific threshold, the wave crest velocity exceeds the wave celerity. Consequently, the crest gradually steepens until it eventually breaks into a series of solitary waves. In the context of WWO, wave breaking is simulated through a local search performed in the vicinity of the wave y*, resulting in a new best solution (where y becomes the new y*). To elaborate further, k dimensions are randomly selected, where k is an integer ranging from 1 to a predefined value k_{max}. At each dimension d, a new wave y' is generated as Eq. (3.12):

$$y'(d) = y(d) + N(0,1).\beta L(d) \tag{3.12}$$

Algorithm 3.3: Algorithm of WWO

Randomly initialize a population P of n waves (solutions)
While the stop criterion is not satisfied, do
for each y in P, do
Propagate y to a new y_0 based on Eq. (3.8)
if f(y) < f'(y_0), then
if f(y*) < f'(y_0), then
Break y_0 based on Eq. (3.12);
Update y* with y_0
Replace y with y_0 ;
else
Decrease y:h by one
if y:h = 0 then
Refract y to a new y_0 based on Eq. (3.9) and (3.10);
Update the wavelengths based on Eq. (3.11);
return y*.

d) Proposed Hybrid APSWWO Algorithm

This proposed system developed a hybrid APSWWO algorithm to optimize the hyperparameters of CNN. The APSO algorithm starts by initializing a population of particles within the search space. Each particle represents a potential solution to the optimization problem. The positions and velocities of the particles are randomly initialized, and each particle's personal best positions (p_{best}) are set to their initial positions. The global best position (g_{best}) is initially set as the best position among all the p_{best} positions. In each iteration, the particles update their velocities and positions based on their current velocities, distances to their p_{best} positions, and distances to the g_{best} position. The velocity update guides the particles toward promising regions of the search space. The new positions are determined by adding the updated velocities to the current positions. After updating the positions, the fitness of each particle's new position is evaluated using the objective function. If a particle's new position has a better fitness than its current p_{best} position, its p_{best} position is updated. The g_{best} position is updated by selecting the best position among all the p_{best} positions. This iterative update process continues until a termination criterion is met, such as reaching a maximum number of iterations or achieving a desired solution quality. A set of best individuals have been found at the end of the algorithm. In WWO, the solution space is analogous to a seabed area, and the fitness of a solution is inversely measured by its seabed depth. The algorithm initializes a population of waves, each representing a potential solution with a specific height and wavelength. The best individuals of APSO are used as initialize wave of WWO. Through a series of iterative steps, including propagation, refraction, and breaking, the waves explore and exploit the search space. Propagation moves the waves towards potentially better solutions, while refraction guides them towards regions of higher fitness. Breaking operation enhances local search. The algorithm iteratively updates the waves until a termination criterion is met, aiming to converge to an optimal solution represented by the wave with the highest fitness. The algorithm of the proposed APSWWO is depicted in Algorithm 3.4. The flowchart of APSWWO is presented in Figure 3.7.

Algorithm 3.4: Algorithm of APSWWO

1. Initialize the swarm of particles for APSO
2. Initialize particle Velocity $v_{p,}$ and position X_p
3. Initialize $p_{best}, and\ g_{best}$ for each particle
4. Set p_{best} for each particle initially as its current position.
5. Set g_{best} as the best position among all the particles in the swarm.
6. Repeat until a termination condition is met.
7. Evaluate the fitness of f_j for each particle in the APSO population using the fitness function.
8. Average fitness is to be calculated for APSO as f_{avg} using Eq. (3.4)
9. Ratio αj is calculated using Eq. (3.5)
10. If $\alpha_j \geq 1\ then$
11. Calculate w_i using Eq. (3.6)
12. Else
13. Calculate w_i using Eq. (3.7)

14. Update the velocity and position of the particles using Eq. (3.2) and (3.3)
15. Find the value of $currp_{best}, currg_{best}$ according to $New\ v_p$ and $New\ X_p$
16. For each particle in population, do
17. If $currp_{best} > p_{best}$ then

$$p_{best} = currp_{best}$$

18. If $currg_{best} > g_{best}$ then

$$g_{best} = currg_{best}$$

19. Select the best individuals f_{besti} for each particle have been processed.
20. Initialize the best individuals of each particle as the waves (y_i) for WWO
21. Initialize y*
22. while the stop criterion is not satisfied, do
23. for each wave y_i do
24. Propagate y_i to a new y_0 based on Eq. (3.8)
25. if $f(y_i) < f'(y_0)$, then
26. if $f(y^*) < f'(y_0)$, then
27. Break y_0 based on Eq. (3.12)
28. Update y* with y_0
29. Replace y_i with y_0
30. else
31. Decrease y_i:h by one
32. if y_i : h = 0 then
33. Refract y_i to a new y_0 based on Eq. (3.9) and (3.10)
34. Update the wavelengths based on Eq. (3.11)
35. return y* (optimal solution)

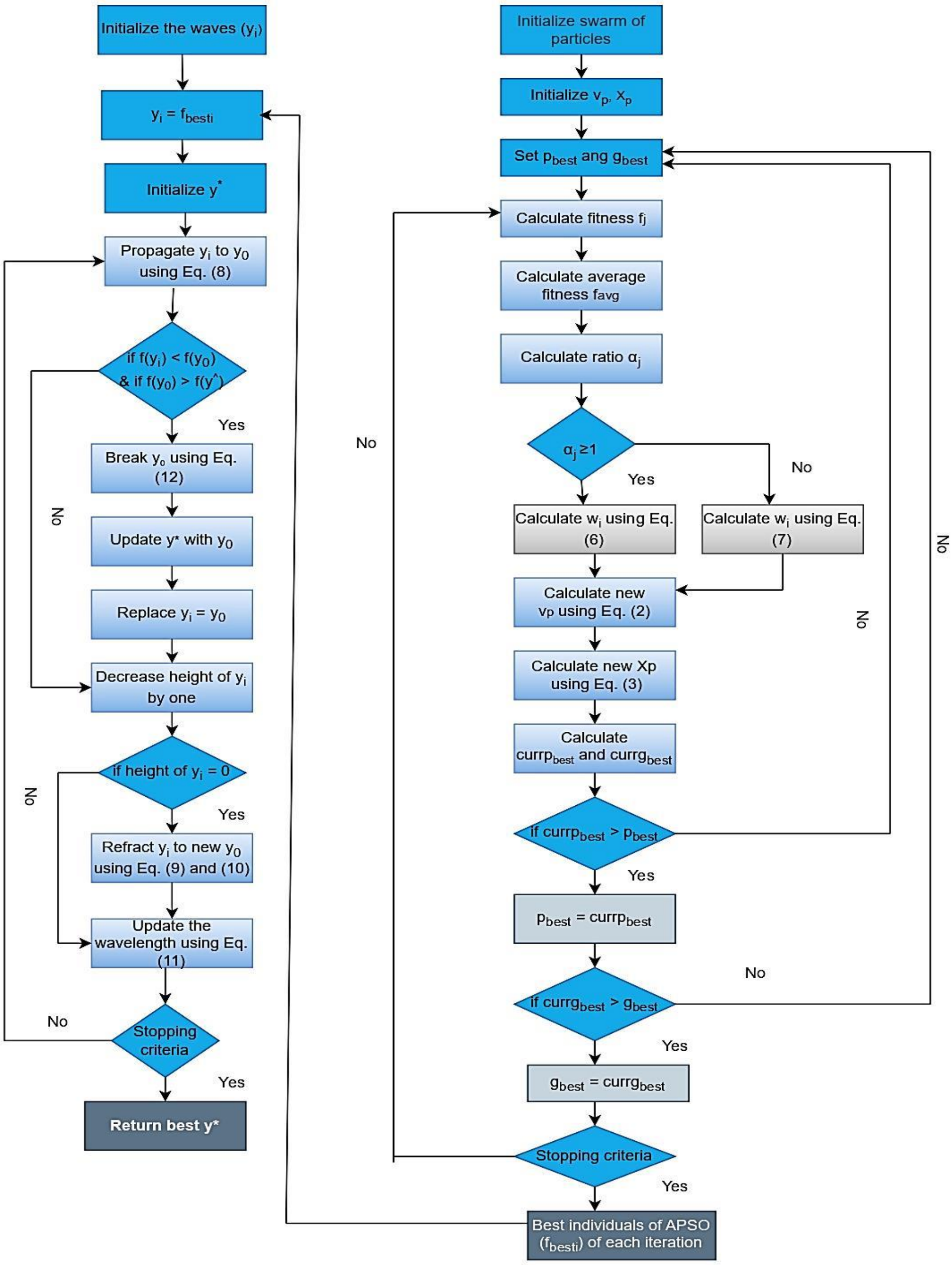


Figure 3.7: Flowchart of proposed APSWWO algorithm

3.2.5 Proposed Optimized ANN Model

This study has introduced a novel hybrid optimization algorithm Adaptive Particle Swarm Water Wave Optimization algorithm to optimize the hyperparameters of our proposed 26-layer Convolutional Neural Network model and 13-layer LiteViT architecture. This comprehensive approach combines elements of Adaptive Particle Swarm Optimization and Water Wave Optimization to fine-tune the models effectively.

In the APSWWO algorithm, the population is composed of combinations of hyperparameters for both the CNN and LiteViT models as shown in Table 3.4. Each particle in the APSWWO population represents a specific configuration for these hyperparameters, allowing for a versatile exploration of the models' parameter spaces. For the 26-layer Convolutional Neural Network model, the particle's hyperparameter combination includes various learning rates, batch sizes, epoch numbers, dropout rates for CNN layers, and dropout rates for the attention mechanisms. These hyperparameters can vary widely to facilitate comprehensive exploration. On the other hand, for the 13-layer LiteViT model, the population's particles encompass learning rates, batch sizes, epoch numbers, number of attention heads, projection dimensions, dropout rates for CNN layers, and dropout rates for attention layers. These parameters provide flexibility to adapt the LiteViT model's structure.

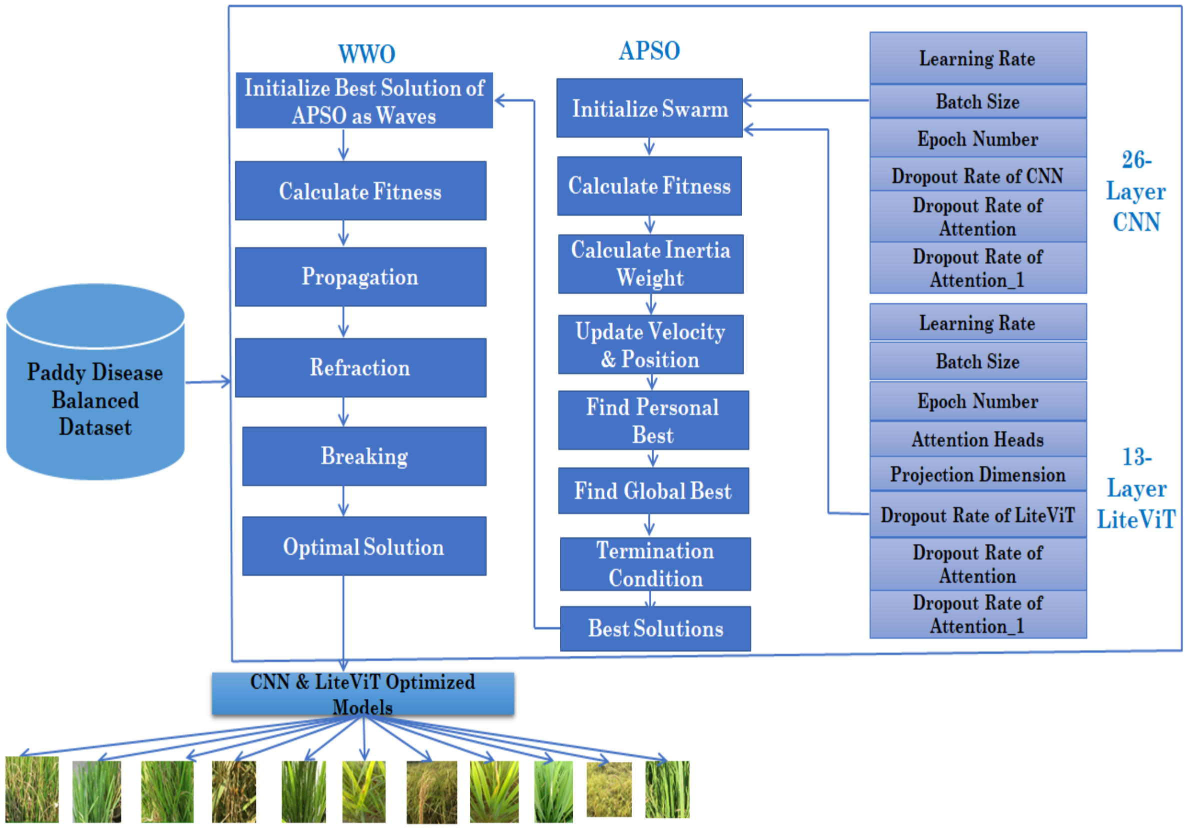


Figure 3.8: Proposed APSWWO approach for CNN and LiteViT optimization

The APSWWO algorithm systematically optimizes these hyperparameters during its iterative process. It initializes a population of particles, where each particle signifies a distinct hyperparameter configuration for both the CNN and LiteViT models. These particles' positions and velocities are randomly assigned, and their personal best positions (p_{best}) are set to their initial configurations. Additionally, a global best position (g_{best}) is initiated as the best among all the p_{best} positions. In the iterative phase of the optimization process, the particles adapt by updating their velocities and positions based on various factors, including their current velocities, distances to their p_{best} positions, and distances to the g_{best} position. The velocity adjustments guide the particles towards promising regions within the hyperparameter search space. Updated positions are determined by incorporating the modified velocities into their current configurations. Following these updates, the fitness of each particle's new hyperparameter set is evaluated using the Eq.(3.13) as the fitness function.

$$\text{Fitness Function} = \frac{\text{Number of correctly classified instances}}{\text{Total number of instances}} \quad (3.13)$$

When a particle's new configuration outperforms its current p_{best} position, the p_{best} is updated. Simultaneously, the g_{best} position evolves by selecting the optimal configuration from all the p_{best} positions. This iterative optimization process continues until a termination criterion is met. The APSO phase stops after 50 iterations to allow for an initial optimization.

Table 3.4: Description of hyper-parameter

Algorithms	**Hyper-parameters**	**Range**
26-Layer CNN	Learning Rate	0.001-0.1
	Batch Size	2-512
	Epoch Number	1-500
	Dropout Rate of CNN	0.1-1.0
	Dropout Rate of Attention Layer	0.1-1.0
	Dropout Rate of Attention_1 Layer	0.1-1.0
13-Layer LiteViT	Learning Rate	0.001-0.1
	Batch Size	2-512
	Epoch Number	1-500
	Attention Heads	2-300
	Projection Dimension	4-512
	Dropout Rate of LiteViT	0.1-1.0
	Dropout Rate of Attention Layer	0.1-1.0
	Dropout Rate of Attention_1 Layer	0.1-1.0

The second phase of propose APSWWO approach focuses on Water Wave Optimization. WWO initiates a population of waves, each wave representing a potential hyperparameter configuration with specific attributes, such as height and wavelength. The best individuals obtained through the APSO algorithm serve as the initial waves of the WWO phase. Through a sequence of iterative steps, including propagation, refraction, and breaking, the waves explore and exploit the hyperparameter search space. The algorithm continually updates the waves until a termination criterion is met, with the ultimate goal of converging to an optimal solution represented by the wave with the highest fitness, aiming for an accuracy of 100%. The graphical representation of proposed APSWWO approach for CNN and LiteViT optimization is depicted in Figure 3.8. This multi-phase approach ensures that the hyperparameters of both the 26-layer CNN model and the 13-layer LiteViT model are tuned systematically, aiming to attain the highest achievable accuracy for each.

3.2.6 Training and Testing

This study has used 80% of the dataset for training the 26-Layer CNN and 13-Layer LiteViT models, and reserving 20% for rigorous testing. To combat overfitting, the study implemented early stopping, which halted training once validation metrics ceased to improve, and dropout layers, which introduced regularization by randomly deactivating a subset of neurons during training.

3.4 Explainable AI Technique in Paddy Disease Detection

Explainable AI Techniques are methods used to understand and explain the predictions and decision-making processes of machine learning models, especially deep neural networks. These techniques are crucial for gaining insights into how a model arrives at its conclusions, identifying important features, and ensuring transparency and trustworthiness in various applications, such as healthcare, finance, and autonomous systems. This study has utilized Gradient-weighted Class Activation Mapping as an Explainable AI technique for both the 26-layer CNN and 13-layer LiteViT models.

Gradient-weighted Class Activation Mapping

Grad-CAM is a technique that provides visual explanations of neural network predictions, highlighting the regions of input data (e.g., images) that contribute the most to the model's decision. It does so by generating a heatmap superimposed on the input data. Grad-CAM works by computing the gradients of the target class with respect to the final convolutional layer of the model. It captures the regions of the input image that are most important for the network's classification, helping to identify what features and areas of an input are relevant for the model's predictions. This visual explanation can be valuable for understanding and validating model behavior, especially in applications where interpretability and trustworthiness are critical, such as medical image analysis, where it can help doctors and researchers comprehend and trust the model's diagnostic decisions. This study has used Grad-CAM to visualize which parts of the paddy disease data (images) the CNN and LiteViT models are focusing on when making their predictions.

3.4 Performance Analysis

This study has used model construction to evaluate the efficacy and usefulness of ANN algorithms for paddy leaf disease prediction. A confusion matrix and all relevant metrics, including the True Positives, True Negatives, False Positives, False Negatives, accuracy, error rate, precision, recall, F1-score, ROC curve, precision-recall curve, and learning curve are used to evaluate a model's performance.

Confusion matrix: One of the most straightforward methods for assessing the model's efficacy and accuracy is the confusion matrix. A table containing the two dimensions, "Actual class" and "Predicted class," in each dimension is the confusion matrix.

True Positives (TP): True positives are the cases when the actual class of the data point is True, and the predicted is also True.

True Negatives (TN): True negatives are the cases when the actual class of the data point is False, and the predicted is also false.
False Positives (FP): False positives are the cases when the actual class of the data point is False, and the predicted is True.
False Negatives (FN): False negatives are the cases when the actual class of the data point is True, and the predicted is False.
Recall: Recall refers to a test's ability to designate an individual with paddy disease as positive. A highly sensitive test means that there are few false negative results, and thus fewer cases of paddy disease are missed. It is also known as the True Positive rate (TPR). It can be calculated using Eq.(3.14).

$$\text{Recall} = \frac{TP}{TP+FN} \tag{3.14}$$

Precision: Precision tells what fraction of predictions of a positive class are actually paddy diseases positive and can be expressed using Eq.(3.15). The high precision means the result of the measurements is consistent or the repeated values of the reading are obtained. The low precision means the value of the measurement varies.

$$\text{Precision} = \frac{TP}{TP+FP} \tag{3.15}$$

F1-score: The harmonic mean of precision and recall is called F1-score as shown in Eq.(3.16). The high F1-score indicates perfect precision and recall of the proposed model.

$$\text{F1-score} = 2*\frac{Recall*Precision}{Recall+Precision} \tag{3.16}$$

Accuracy: Accuracy is calculated as the number of all correct predictions of paddy disease divided by the total number of the dataset using Eq.(3.17).

$$\text{Accuracy} = \frac{TP+TN}{TP+FP+TN+FN} \tag{3.17}$$

Error Rate: The error rate, also known as the classification error rate or misclassification rate, is a metric used to evaluate the performance of a classification model. It represents the proportion of incorrectly classified instances in a dataset. The error rate can be calculated using the Eq.(3.18):

Error Rate = (Number of Misclassified Instances) / (Total Number of Instances) (3.18)

ROC Curve: ROC (Receiver Operating Characteristics) is a metric for evaluating classification models, plotting True Positive Rate (TPR) against False Positive Rate (FPR). The ROC-AUC (Area Under the ROC Curve) score measures overall model performance, with higher values indicating better classification. A score of 0.5 represents random guessing, while 1 signifies perfect classification. Thus, a higher ROC-AUC value suggests a more effective classifier.
Precision-Recall Curve: The Precision-Recall curve is an evaluation tool for paddy disease classification, focusing on the trade-off between precision (positive predictive value) and recall (true positive rate) at different decision thresholds. It helps assess a model's ability to correctly classify positive instances while minimizing false positives. The Average Precision

(AP) value summarizes the Precision-Recall curve, providing a single number to measure a model's overall performance.

Learning Curve: A learning curve is a visual representation of how a machine learning model's training and testing loss changes over time (typically across epochs or iterations). It helps analyze the model's performance and identify issues like overfitting or underfitting. The training loss measures how well the model fits the training data, while the testing loss assesses its generalization to unseen data.

3.5 Summary

This study aims to enhance paddy disease prediction through optimized Artificial Neural Network (ANN) models using a new hybrid optimization algorithm. The methodology entails different key stages: image acquisition, preprocessing, dataset balancing, development of hybrid optimization methods, ANN utilization, integration of model interpretability techniques, and thorough performance evaluation. This section provides a thorough examination of the novel hybrid optimization algorithm's function in optimizing ANN hyperparameters for paddy disease prediction, offering a comprehensive overview of the research methodology.

CHAPTER 4

RESULTS & DISCUSSION

4.1 Introduction

The main objective of this study is to develop a model using Artificial Neural Network (ANN) to accurately detect paddy diseases at an early stage. Additionally, the aim is to devise a novel hybrid optimization algorithm, APSWWO, to efficiently optimize the parameters of the ANN. This will enable farmers and agricultural professionals to take immediate action and implement timely disease management strategies. The study also involves conducting a comprehensive analysis among different neural network-based models and optimization algorithms to identify the most effective ones. Furthermore, the implementation of Explainable AI techniques, such as Grad-CAM, will provide detailed insights into the decision-making processes of the ANN-based paddy disease detection system, facilitating better understanding and interpretation of the inner workings of these models.

4.2 Paddy Disease Detection using ANN

In this study, the performance analysis of 26-layer CNN (CNN_Original, CNN_PSO, CNN_APSO, CNN_WWO, and CNN_APSWWO) and 13-layer LiteViT (LiteViT_Original, LiteViT_PSO, LiteViT_APSO, LiteViT_WWO, and LiteViT_APSWWO) was evaluated. The comparative analysis of each ANN model was conducted in its original, single optimization, and hybrid optimization forms.

4.2.1 Convolutional Neural Network

In this section, the performance analysis and interpretability of CNN_Original, CNN_PSO, CNN_APSO, CNN_WWO, and CNN_APSWWO have been presented.

a) CNN

This sub-section delves into the training and testing outcomes of the proposed 26-layer CNN_Original model, utilizing various performance metrics. A comprehensive evaluation of the training results of CNN_Original is presented, providing in-depth insights into its performance. The propose 26-layer CNN model, undergoes evaluation without the application of any optimization algorithm. The key hyperparameters, such as the learning rate, batch size, epoch number, dropout rates for both the CNN model and the attention layers (attention and attention_1) of this model, are set to specific values: 0.001, 16, 100, 0.5, 0.2, and 0.5, respectively. Figure 4.1 illustrates the confusion matrix generated for this model. In this matrix, correctly predicted instances, denoted by blue cells, match the model's output with the target values, while white cells indicate misclassified instances. The model accurately classified 99 instances of BLB, 107 instances of BPH, 108 instances of brown spot, 115 instances of false smut, 115 instances of healthy plants, 113 instances of hispa, 79 instances of leaf scaled, 68 instances of neck blast, 93 instances of tungro, 74 instances of sheath blight rot, and 107 instances of stemborer. However, there were misclassifications, with 3 instances incorrectly classified as BLB, 0 as BPH, 18 as brown spot, 1 as false smut, 6 as healthy plants, 0 as hispa, 22 as leaf scaled, 117 as neck blast, 25 as tungro, 26 as sheath blight rot, and none as stemborer.

In Figure 4.2, the ROC curve of CNN_Original is presented, depicting the trade-off between TPR and FPR across varying classification thresholds. Each point on this curve corresponds to a specific threshold value. Notably, the ROC-AUC scores for BLB, BPH, brown spot, false smut, healthy plant, hispa, leaf scaled, neck blast, tungro, sheath blight rot, and stemborer are exceptional, with values of 1.00, 1.00, 0.98, 1.00, 1.00, 0.98, 0.96, 0.98, 0.98, 0.98, and 1.00, respectively. This highlights the model's outstanding capability in accurately identifying these particular diseases, with BLB, BPH, false smut, healthy plant, and stemborer achieving the highest ROC-AUC scores, as demonstrated in Figure 4.5(d), while other classes also exhibit strong performance.

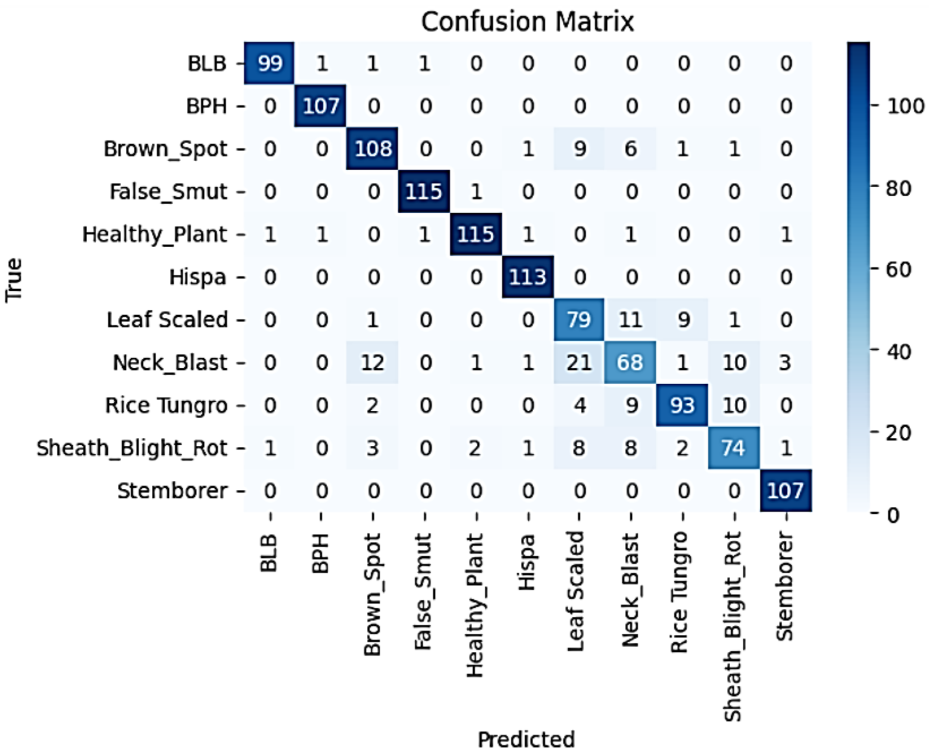


Figure 4.1: Confusion matrix of CNN_Original

Figure 4.3 illustrates the precision-recall curves for CNN_Original in the context of paddy disease classification. These curves provide insights into the balance between precision (the accuracy of positive predictions) and recall (the ability to identify all positive cases) as the classification threshold undergoes variations. Notably, the Average Precision scores for BLB, BPH, brown spot, false smut, healthy plant, hispa, leaf scaled, neck blast, tungro, sheath blight rot, and stemborer are exceptional, with values of 1.00, 1.00, 0.93, 1.00, 0.99, 1.00, 0.81, 0.74, 0.90, 0.87, and 1.00, respectively. As demonstrated in Figure 4.5(e), BLB, BPH, false smut, healthy plant, and stemborer attain the highest AP scores, while other classes also exhibit strong performance. Importantly, AP values of 1.00 for BLB, BPH, false smut, and stemborer indicate the CNN_Original's ability to effectively classify positive instances while minimizing false positives in these classes. Conversely, the lowest AP score is observed for the neck blast class, as shown in Figure 4.5(e).

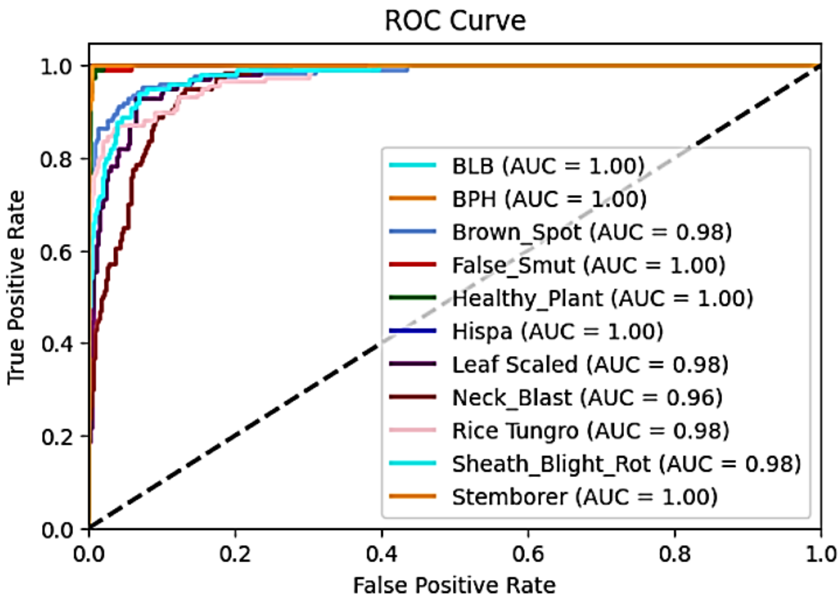


Figure 4.2: ROC curve of CNN_Original

Figure 4.4 displays the learning curve of CNN_Original. Early stopping was implemented, limiting the model to 23 epochs. During this training, the training loss consistently decreased with each epoch, ultimately reaching 0.1884. However, the testing loss exhibited fluctuating behavior throughout the epochs, resulting in a final testing loss of 1.103.

Table 4.1 provides the classification results for CNN_Original, depicting various performance metrics for each class. In the BLB class, recall, precision, F1-score, ROC_AUC score, and AP value stand at 98%, 97%, 98%, 1.00, and 1.00, respectively. For the BPH class, these metrics are 98%, 100%, 99%, 1.00, and 1.00. The brown spot class exhibits values of 85%, 86%, 85%, 0.98, and 0.93 for recall, precision, F1-score, ROC_AUC score, and AP value, respectively. In the false smut class, they reach 98%, 99%, 99%, 1.00, and 1.00. For the healthy plant class, these values are 97%, 95%, 96%, 1.00, and 0.99, while the Hispa class shows 97%, 100%, 98%, 0.98, and 1.00. Leaf scaled has a recall of 65%, precision of 78%, F1-score of 71%, ROC_AUC score of 0.96, and AP value of 0.81. Neck blast class records a recall of 66%, precision of 58%, F1-score of 62%, ROC_AUC score of 0.98, and AP value of 0.74. Tungro class achieves 88% recall, 79% precision, 83% F1-score, ROC_AUC score of 0.98, and AP value of 0.90, while sheath blight rot class displays a recall of 77%, precision of 74%, F1-score of 76%, ROC_AUC score of 0.98, and AP value of 0.87. Lastly, the stemborer class exhibits a recall of 96%, precision of 100%, F1-score of 98%, ROC_AUC score of 1.00, and AP value of 1.00. Figure 13 visually presents the performance metrics for each class using CNN_Original, where BPH and false smut outshine others with the highest recall of 98%, while leaf scaled attains the lowest recall of 66%. BPH, hispa, and stemborer lead in precision with a perfect score of 100%, while neck blast trails with the lowest recall at 58%. The top F1-score of 99% is achieved by BPH and false smut, while neck blast lags behind with a 62% F1-score for the CNN_Original model.

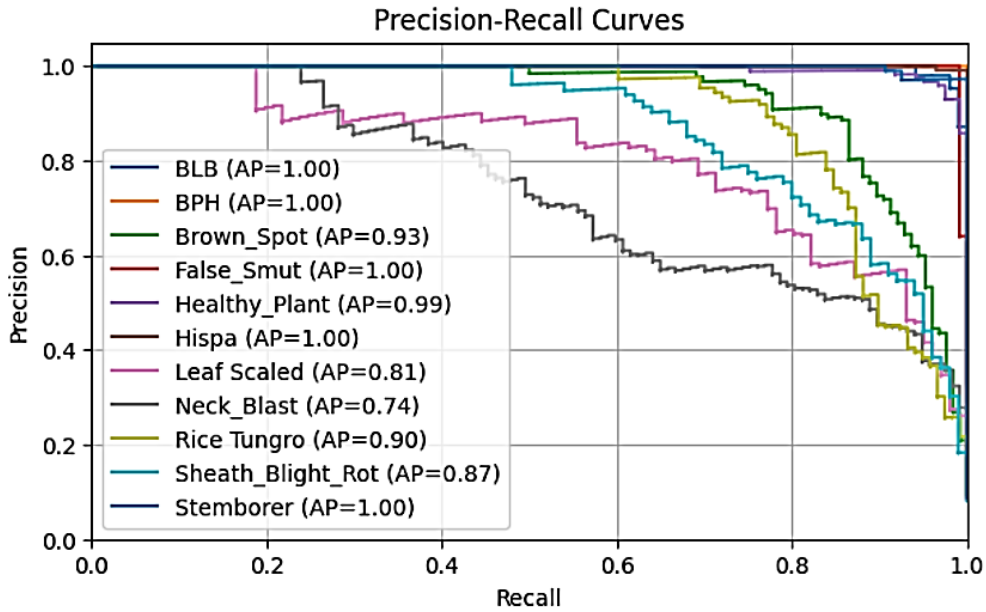


Figure 4.3: Precision-Recall curves of CNN_Original

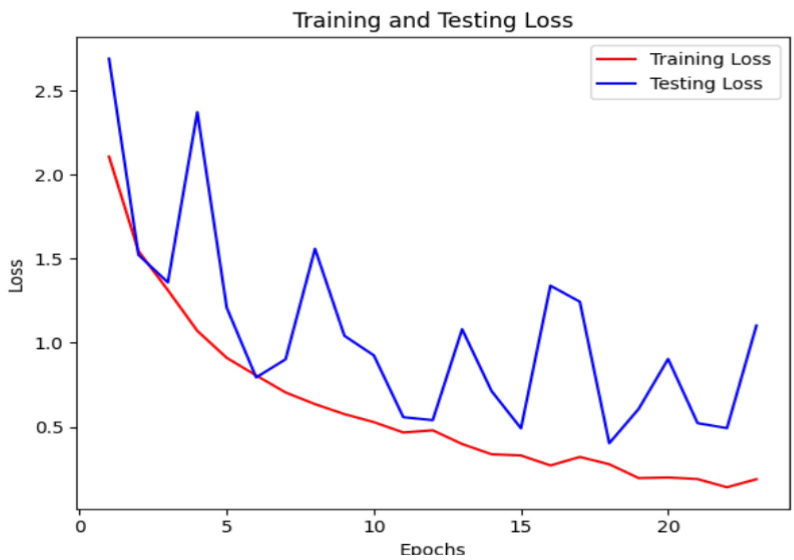
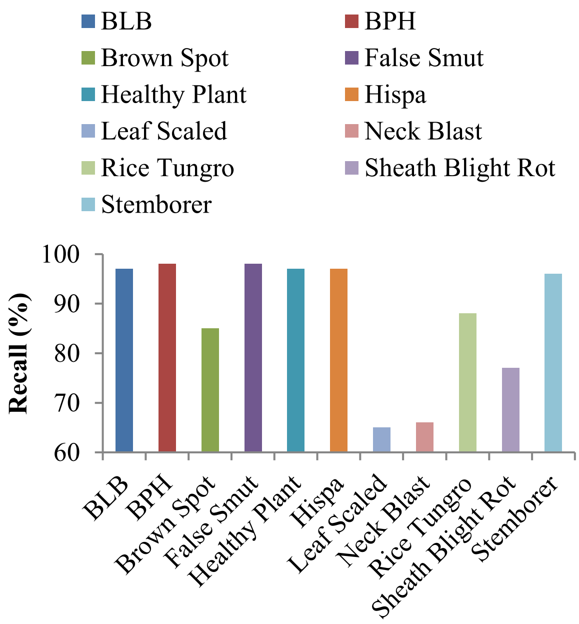


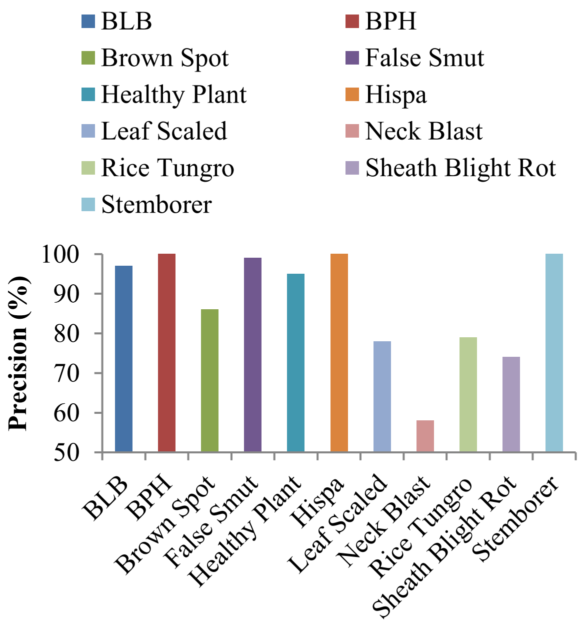
Figure 4.4: Learning curve of CNN_Original

Table 4.1: Performance table of CNN_Original

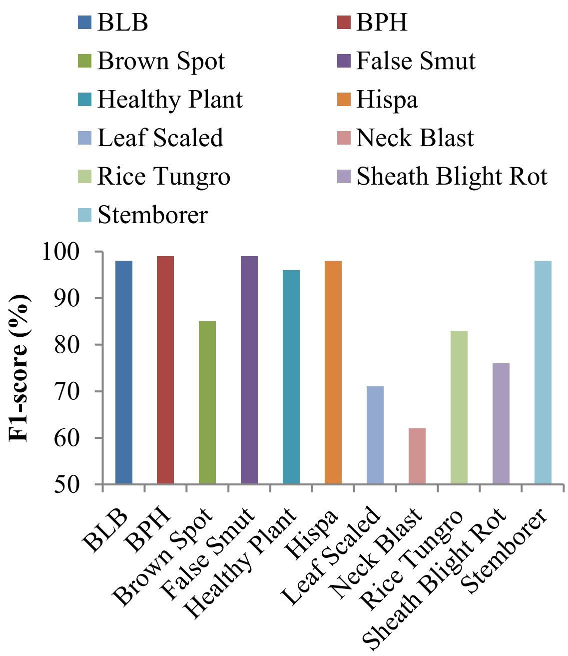
Class	Correctly Classified Instance	Incorrectly Classified Instance	Recall (%)	Precision (%)	F1-score (%)	ROC-AUC score	AP value
BLB	99	3	97	97	98	1.00	1.00
BPH	107	0	98	100	99	1.00	1.00
Brown Spot	108	18	85	86	85	0.98	0.93
False Smut	115	1	98	99	99	1.00	1.00
Healthy Plant	115	6	97	95	96	1.00	0.99
Hispa	113	0	97	100	98	0.98	1.00
Leaf Scaled	79	22	65	78	71	0.96	0.81
Neck Blast	68	117	66	58	62	0.98	0.74
Paddy Tungro	93	25	88	79	83	0.98	0.90
Sheath Blight Rot	74	26	77	74	76	0.98	0.87
Stemborer	107	0	96	100	98	1.00	1.00
Average	98	19.82	87.73	87.82	87.73	0.983	0.931



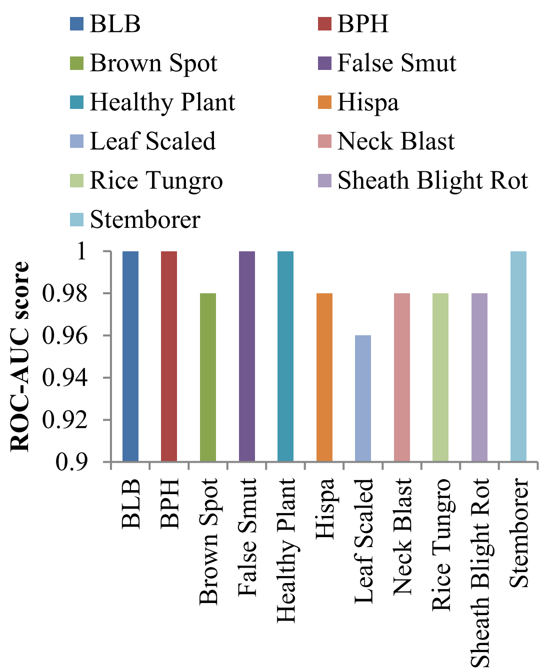
(a) Recall



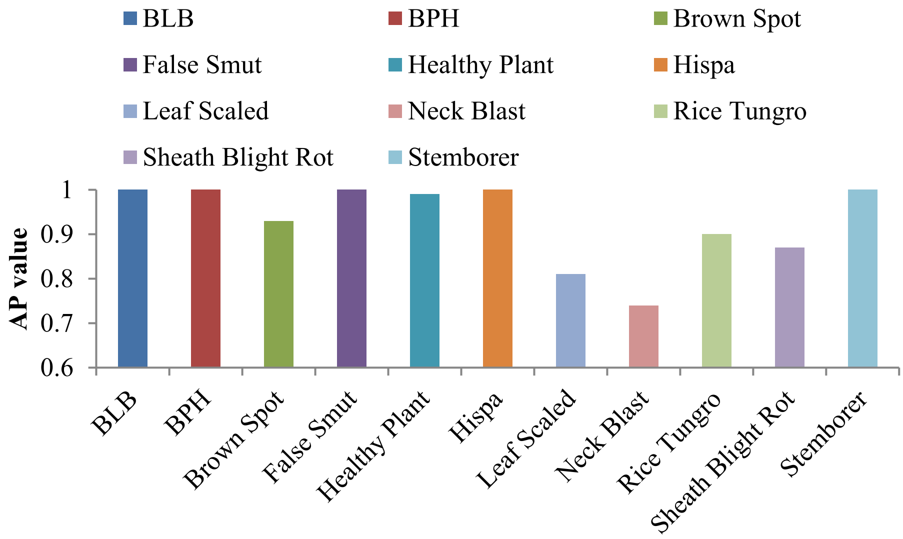
(b) Precision



(c) F1-score



(d) ROC-AUC score



(e) Average Precision value

Figure 4.5: Performance metrics for each class using CNN_Original

b) CNN_PSO

A thorough evaluation of the training outcomes for CNN_PSO is presented, offering an in-depth analysis of its performance. The proposed 26-layer CNN model underwent evaluation employing PSO algorithms, leading to the determination of crucial hyperparameters. Specifically, the hyperparameter values for the 26-layer CNN model using PSO are as follows: a learning rate of 0.004, a batch size of 147, an epoch number of 64, and dropout rates of 0.4 for the CNN model, 0.4 for the attention layers, and 0.5 for the attention_1 layer within this model. Figure 4.6 visually represents the confusion matrix generated for this model. Within this matrix, correctly predicted instances, depicted by blue cells, align the model's output with the target values, while white cells signify instances that were misclassified. The model achieved precise classification for 102 instances of BLB, 107 instances of BPH, 103 instances of brown spot, 114 instances of false smut, 113 instances of healthy plants, 113 instances of hispa, 78 instances of leaf scaled, 86 instances of neck blast, 106 instances of tungro, 79 instances of sheath blight rot, and 106 instances of stemborer. However, misclassifications occurred, with 0 instances incorrectly classified as BLB, 0 as BPH, 23 as brown spot, 2 as false smut, 8 as healthy plants, 0 as hispa, 23 as leaf scaled, 31 as neck blast, 12 as tungro, 21 as sheath blight rot, and 1 as stemborer.

In Figure 4.7, the ROC curve of CNN_PSO is showcased, illustrating the trade-off between recall and False Positive Rate across various classification thresholds. Each point on this curve signifies a specific threshold value. Notably, the ROC-AUC scores for BLB, BPH, brown spot, false smut, healthy plant, hispa, leaf scaled, neck blast, tungro, sheath blight rot, and stemborer are outstanding, with values of 1.00, 1.00, 0.97, 1.00, 1.00, 1.00, 0.98, 0.97, 0.99, 0.98, and 1.00, respectively. This underscores the model's exceptional ability to accurately detect these specific diseases. BLB, BPH, false smut, healthy plant, and stemborer achieve the highest ROC-AUC scores, as depicted in Figure 4.10(d), while other classes also demonstrate strong performance.

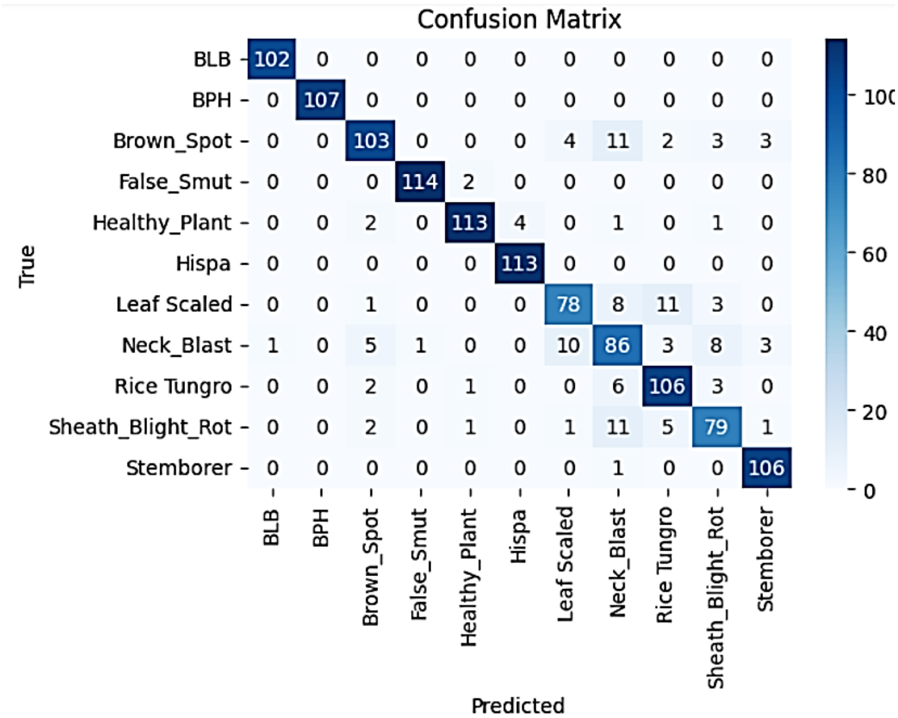


Figure 4.6: Confusion matrix of CNN_PSO

In Figure 4.8, the precision-recall curves for CNN_PSO in paddy disease classification offer valuable insights into the trade-off between precision (accuracy of positive predictions) and recall (ability to identify all positive cases) across varying classification thresholds.

Exceptional Average Precision scores are observed for BLB, BPH, brown spot, false smut, healthy plant, hispa, leaf scaled, neck blast, tungro, sheath blight rot, and stemborer, with values of 1.00, 1.00, 0.91, 1.00, 0.99, 1.00, 0.88, 0.75, 0.93, 0.87, and 1.00, respectively. These scores are prominently highlighted in Figure 4.11(e), where BLB, BPH, false smut, hispa, and stemborer demonstrate superior AP scores. These high AP values signify CNN_PSO's effective classification of positive instances while minimizing false positives in these classes. Conversely, the neck blast class exhibits the lowest AP score, as depicted in Figure 4.10(e).

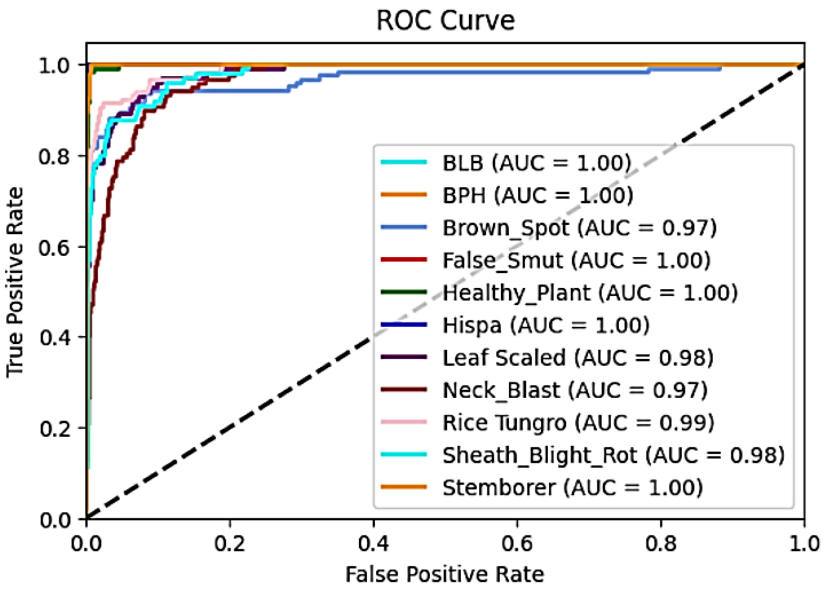


Figure 4.7: ROC curve of CNN_PSO

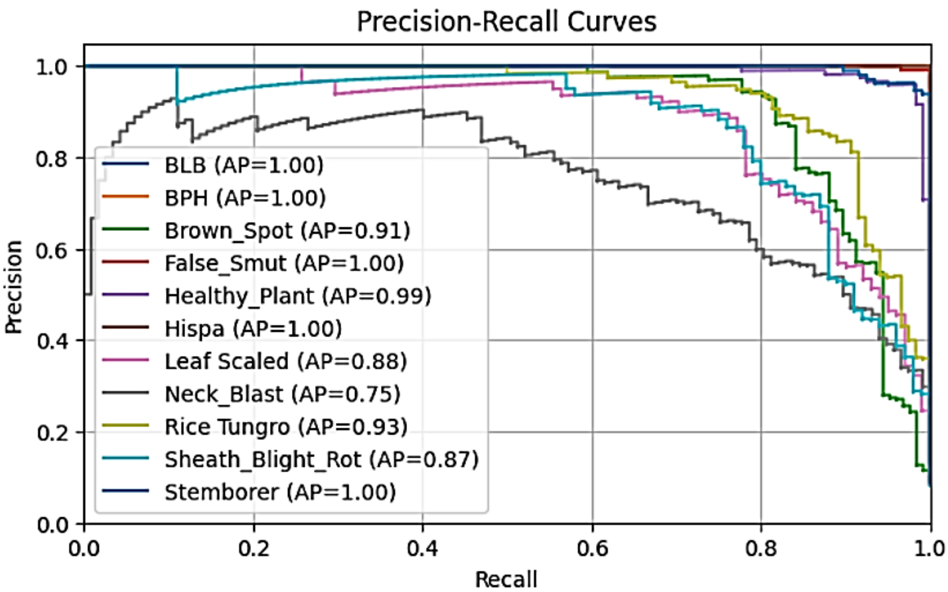


Figure 4.8: Precision-Recall curves of CNN_PSO

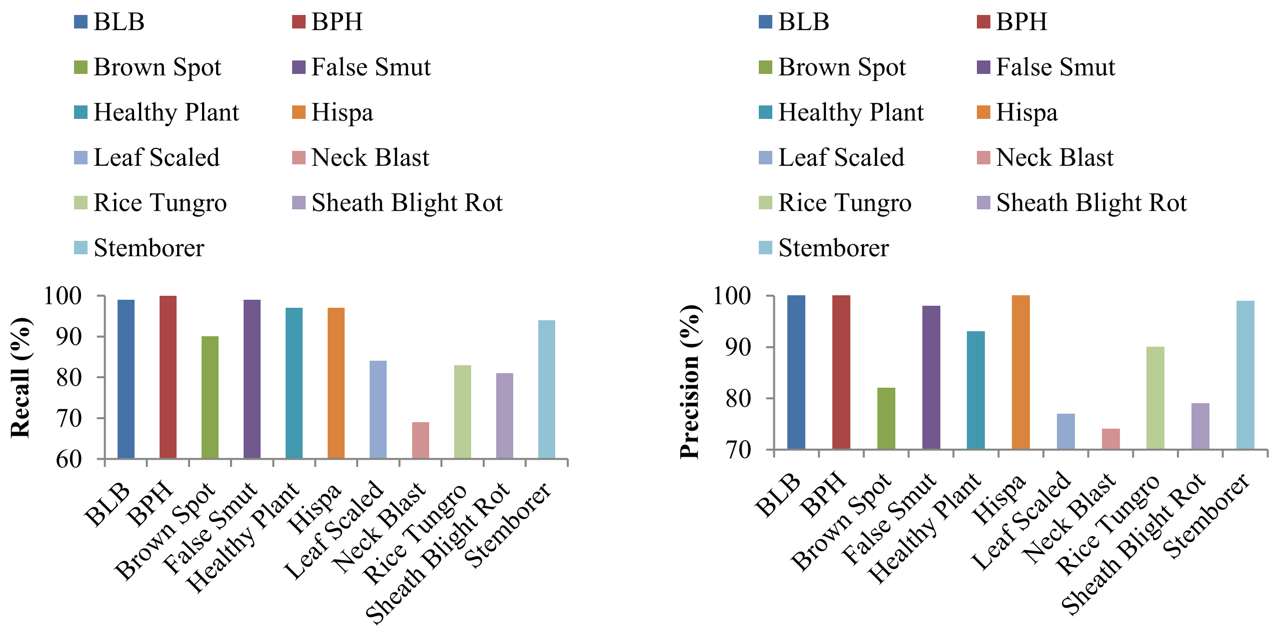
Figure 4.9 showcases CNN_PSO's learning curve, wherein early stopping restricted the training to just 8 epochs. Throughout this training process, the training loss consistently decreased with each epoch, eventually reaching a value of 0.0766. However, in contrast, the testing loss displayed fluctuating behavior across the epochs, culminating in a final testing loss of 0.6784.

Figure 4.9: Learning curve of CNN_PSO

Table 4.2 provides a comprehensive overview of the classification results obtained from CNN_PSO, encompassing various performance metrics for each disease class. In the BLB class, recall, precision, F1-score, ROC_AUC score, and AP value stand at 99%, 100%, 100%, 1.00, and 1.00, respectively. The BPH class exhibits remarkable performance with identical metrics at 100%. In contrast, the brown spot class achieves slightly lower metrics, with a recall of 90%, precision of 82%, F1-score of 85%, ROC_AUC score of 0.97, and AP value of 0.91. The false smut class excels with a recall of 99%, precision of 98%, F1-score of 99%, ROC_AUC score of 1.00, and AP value of 1.00. The healthy plant class maintains strong performance, achieving values of 97% recall, 93% precision, 95% F1-score, ROC_AUC score of 1.00, and AP value of 0.99. The Hispa class also stands out with 97% recall, 100% precision, 98% F1-score, ROC_AUC score of 1.00, and AP value of 1.00. Leaf scaled exhibits a recall of 84%, precision of 77%, F1-score of 80%, ROC_AUC score of 0.98, and AP value of 0.88, while the neck blast class records metrics of 69% recall, 74% precision, 71% F1-score, ROC_AUC score of 0.97, and AP value of 0.75. Tungro class showcases a recall of 83%, precision of 90%, F1-score of 87%, ROC_AUC score of 0.99, and AP value of 0.93, while sheath blight rot class attains a recall of 81%, precision of 79%, F1-score of 80%, ROC_AUC score of 0.98, and AP value of 0.87. Lastly, the stemborer class demonstrates strong performance with a recall of 94%, precision of 99%, F1-score of 96%, ROC_AUC score of 1.00, and AP value of 1.00. Figure 4.10 visually depicts these outstanding performance metrics for each class using CNN_PSO, with BPH achieving the highest recall at 100% and Neck Blast displaying the lowest recall at 69%. BLB, BPH, and Hispa lead in precision, each with a perfect score of 100%, while Neck Blast trails with the lowest recall at 74%. BLB and BPH shine with a top F1-score of 100%, while Neck Blast lags behind with a 71% F1-score for the CNN_PSO model. Remarkably, the BLB class exhibits perfection across all metrics, including recall, precision, F1-score, ROC_AUC score, and AP value.

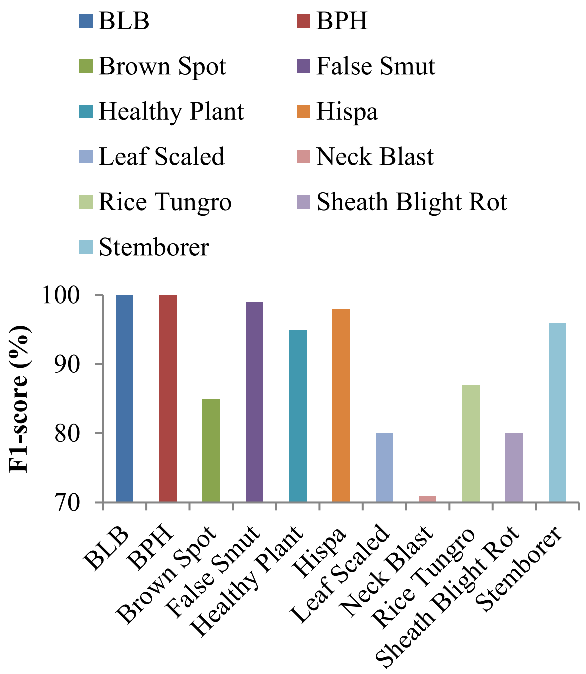
Table 4.2: Performance table of CNN_PSO

Class	**Correctly Classified Instance**	**Incorrectly Classified Instance**	**Recall (%)**	**Precision (%)**	**F1-score (%)**	**ROC-AUC score**	**AP value**
BLB	102	0	99	100	100	1.00	1.00
BPH	107	0	100	100	100	1.00	1.00
Brown Spot	103	23	90	82	85	0.97	0.91
False Smut	114	2	99	98	99	1.00	1.00
Healthy Plant	113	8	97	93	95	1.00	0.99
Hispa	113	0	97	100	98	1.00	1.00
Leaf Scaled	78	23	84	77	80	0.98	0.88
Neck Blast	86	31	69	74	71	0.97	0.75
Paddy Tungro	106	12	83	90	87	0.99	0.93
Sheath Blight Rot	79	21	81	79	80	0.98	0.87
Stemborer	106	1	94	99	96	1.00	1.00
Average	100.64	11	90.27	90.18	90.09	0.99	0.939

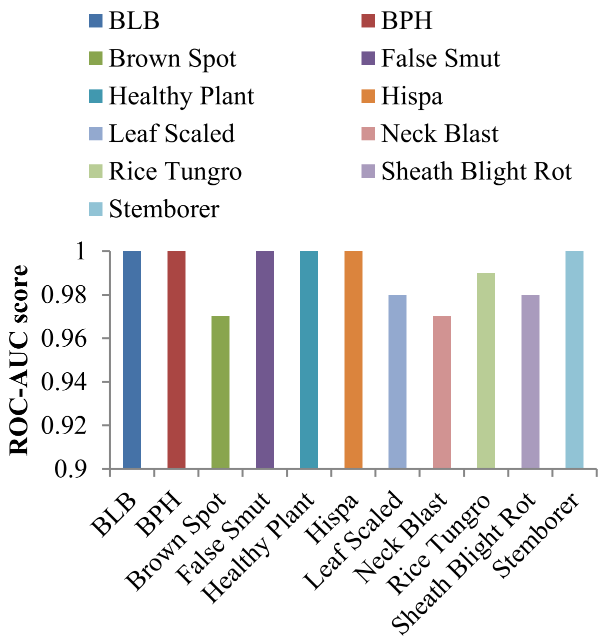


(a) Recall

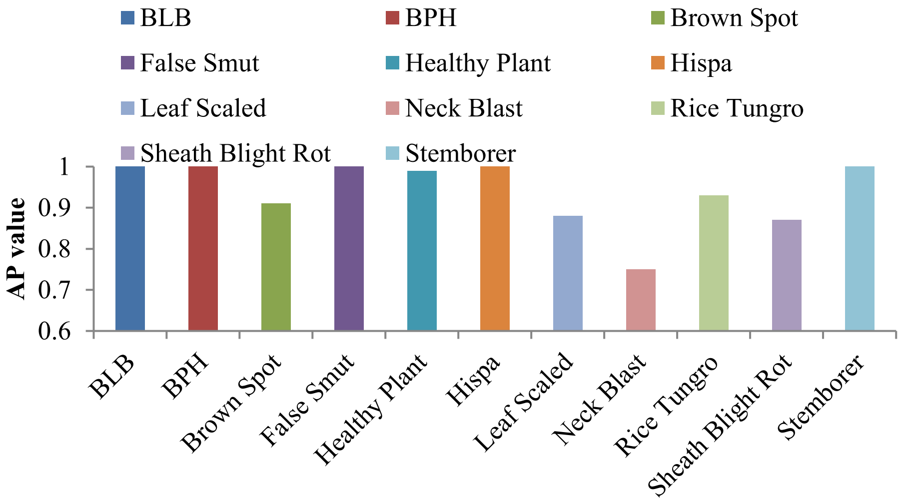
(b) Precision



(c) F1-score



(d) ROC-AUC score



(e) Average Precision value

Figure 4.10: Performance metrics for each class using CNN_PSO

c) CNN_APSO

In this section, the study examines the results of the CNN_APSO model, employing a range of performance metrics for evaluation. A comprehensive evaluation of the training results for CNN_APSO is presented here, providing an in-depth analysis of its performance. The proposed 26-layer CNN model underwent rigorous evaluation using APSO algorithms, resulting in the determination of crucial hyperparameters. Specifically, the hyperparameter values for the 26-layer CNN model using APSO are as follows: a learning rate of 0.001, a batch size of 168, an epoch number of 99, and dropout rates of 0.5 for the CNN model, 0.2

for the attention layers, and 0.5 for the attention_1 layer within this model. Figure 4.11 visually depicts the confusion matrix generated for this model. Within this matrix, correctly predicted instances, highlighted by blue cells, align the model's output with the target values, while white cells represent instances that were misclassified. The model achieved precise classification for 100 instances of BLB, 107 instances of BPH, 113 instances of brown spot, 115 instances of false smut, 114 instances of healthy plants, 113 instances of hispa, 92 instances of leaf scaled, 85 instances of neck blast, 104 instances of tungro, 75 instances of sheath blight rot, and 107 instances of stemborer. However, misclassifications occurred, with 2 instances incorrectly classified as BLB, 0 as BPH, 13 as brown spot, 1 as false smut, 7 as healthy plants, 0 as hispa, 9 as leaf scaled, 32 as neck blast, 14 as tungro, 25 as sheath blight rot, and 0 as stemborer.

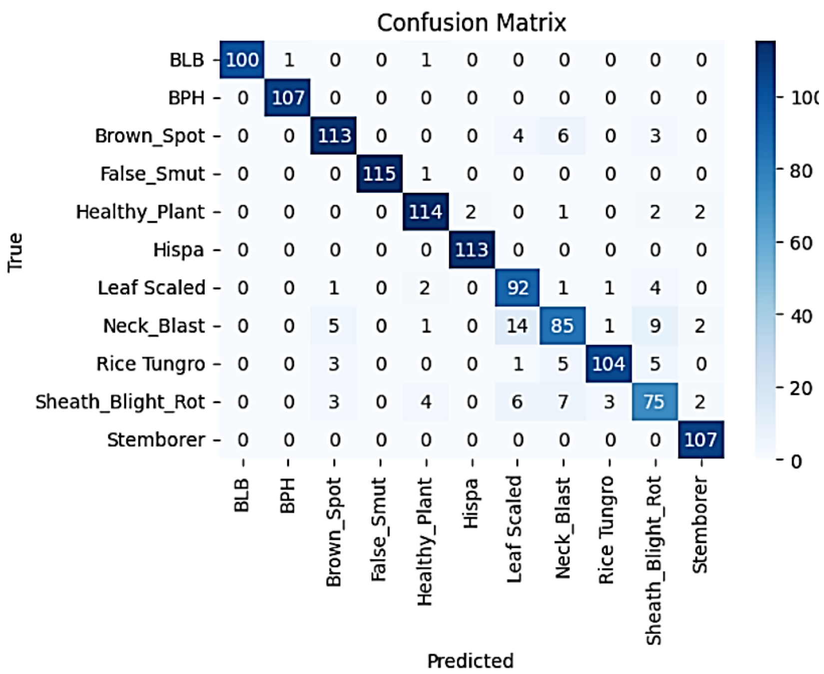


Figure 4.11: Confusion matrix of CNN_APSO

Figure 4.12 provides a visual representation of the ROC curve for CNN_APSO, which illustrates the trade-off between recall and the False Positive Rate at different classification thresholds. Each point on this curve corresponds to a specific threshold value. Notably, the ROC-AUC scores for BLB, BPH, brown spot, false smut, healthy plant, hispa, leaf scaled, neck blast, tungro, sheath blight rot, and stemborer are outstanding, with values of 1.00, 1.00, 0.99, 1.00, 1.00, 1.00, 0.99, 0.98, 0.99, 0.98, and 1.00, respectively. These exceptional scores underscore the model's remarkable ability to accurately detect these specific diseases. BLB, BPH, false smut, healthy plant, hispa, and stemborer achieve the highest ROC-AUC scores, as depicted in Figure 4.15 (d), while other classes also demonstrate strong performance.

Figure 4.13 provides an insightful view of the precision-recall curves for CNN_APSO in the context of paddy disease classification. These curves allow us to understand the trade-off between precision (the accuracy of positive predictions) and recall (the ability to identify all positive cases) as classification thresholds vary. Remarkably, the Average Precision (AP) scores for BLB, BPH, brown spot, false smut, healthy plant, hispa, leaf scaled, neck blast, tungro, sheath blight rot, and stemborer are exceptional, with values of 1.00, 1.00, 0.95, 1.00, 0.99, 1.00, 0.93, 0.85, 0.97, 0.87, and 1.00, respectively. These outstanding AP scores are highlighting the superior performance of CNN_APSO in effectively classifying positive instances while minimizing false positives in these classes. Conversely, the neck blast class exhibits the lowest AP score, as depicted in Figure 4.18 (e).

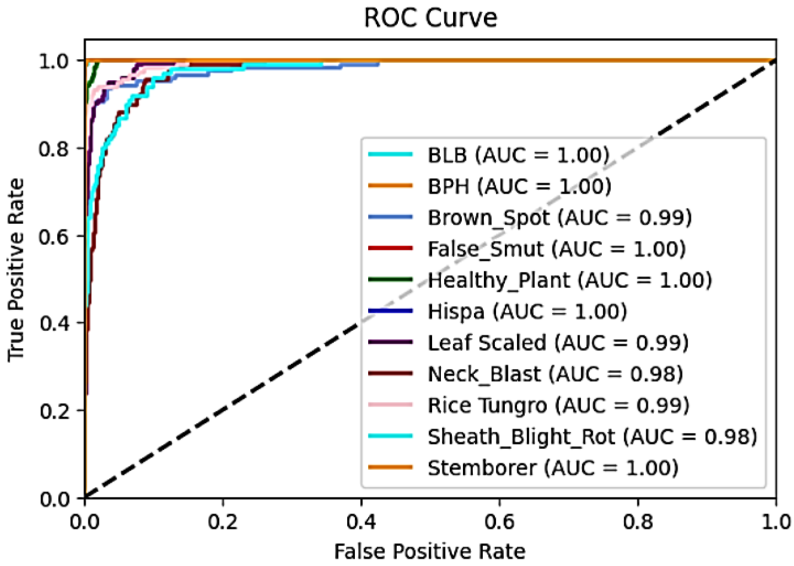


Figure 4.12: ROC curve of CNN_APSO

Figure 4.14 visually represents the learning curve of CNN_APSO. During the training process, early stopping was implemented, limiting the model to just 20 epochs. The training loss consistently decreased with each epoch, reaching a final value of 0.023, indicating effective learning from the training data. However, the testing loss exhibited fluctuations throughout the epochs, resulting in a final testing loss of 0.4936. This suggests that while the model performed well on the training data, it may have encountered some challenges when generalizing to unseen data.

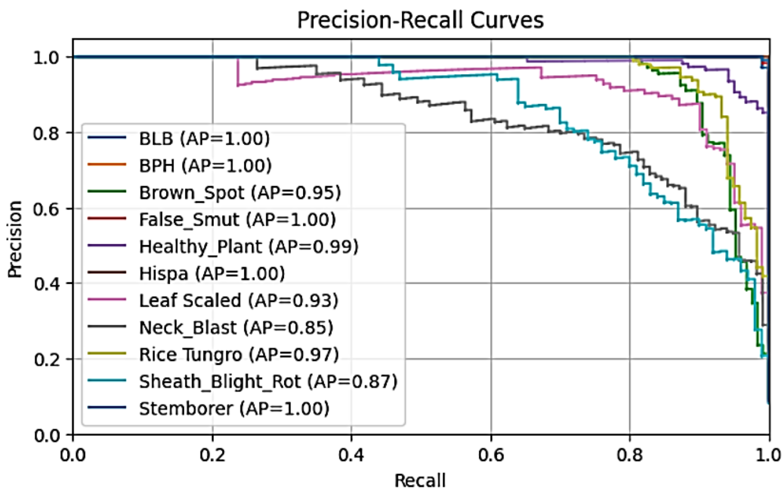


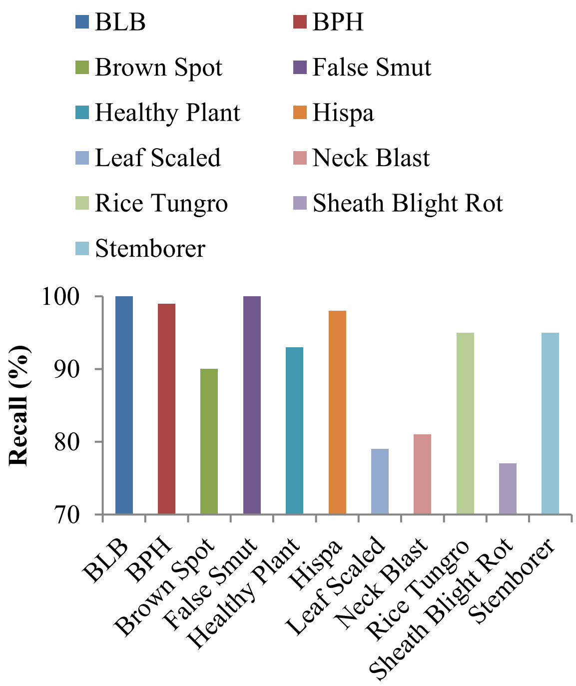
Figure 4.13: Precision-Recall curves of CNN_APSO

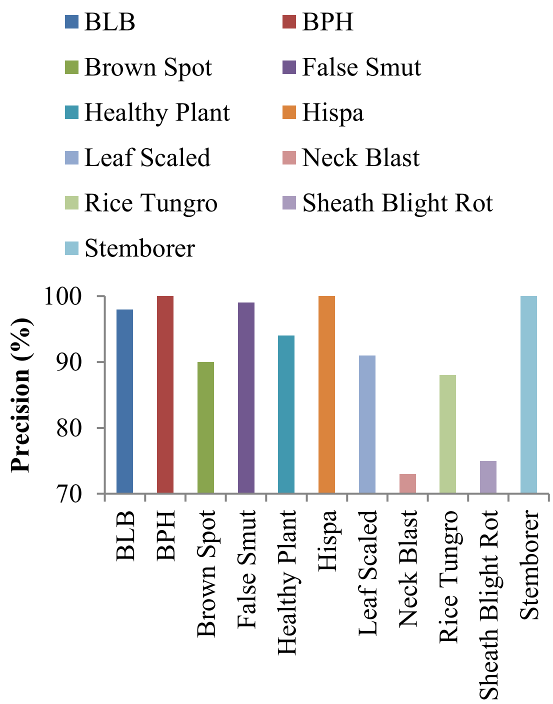
Figure 4.14: Learning curve of CNN_APSO

Table 4.3 presents a comprehensive overview of the classification results obtained from CNN_APSO, showcasing various performance metrics for each disease class. In the BLB class, the model achieves a recall, precision, F1-score, ROC_AUC score, and AP value of 100%, 98%, 98%, 1.00, and 1.00, respectively. Similarly, for the BPH class, these metrics stand at 99%, 100%, 100%, 1.00, and 1.00, respectively. The brown spot class exhibits slightly lower metrics with a recall of 90%, precision of 90%, F1-score of 90%, ROC_AUC score of 0.99, and AP value of 0.95. Meanwhile, the false smut class excels with a recall of 100%, precision of 99%, F1-score of 100%, ROC_AUC score of 1.00, and AP value of 1.00. The healthy plant class maintains strong performance, achieving values of 93% recall, 94% precision, 93% F1-score, ROC_AUC score of 1.00, and AP value of 0.99. Similarly, the Hispa class stands out with 98% recall, 100% precision, 99% F1-score, ROC_AUC score of 1.00, and AP value of 1.00. Leaf scaled exhibits a recall of 79%, precision of 91%, F1-score of 84%, ROC_AUC score of 0.99, and AP value of 0.93, while the neck blast class records metrics of 81% recall, 73% precision, 77% F1-score, ROC_AUC score of 0.98, and AP value of 0.85. The tungro class showcases a recall of 95%, precision of 88%, F1-score of 92%, ROC_AUC score of 0.99, and AP value of 0.97, while sheath blight rot class attains a recall of 77%, precision of 75%, F1-score of 76%, ROC_AUC score of 0.98, and AP value of 0.87. Finally, the stemborer class demonstrates strong performance with a recall of 95%, precision of 100%, F1-score of 97%, ROC_AUC score of 1.00, and AP value of 1.00. Figure 4.15 visually illustrates these exceptional performance metrics for each class using CNN_APSO, with BLB and false smut achieving the highest recall at 100%, while sheath blight rot displays the lowest recall at 77%. BPH and Hispa lead in precision, each with a perfect score of 100%, while neck blast trails with the lowest recall at 73%. BPH and false smut shine with a top F1-score of 100%, while sheath blight rot lags behind with a 76% F1-score for the CNN_APSO model.

Table 4.3: Performance table of CNN_APSO

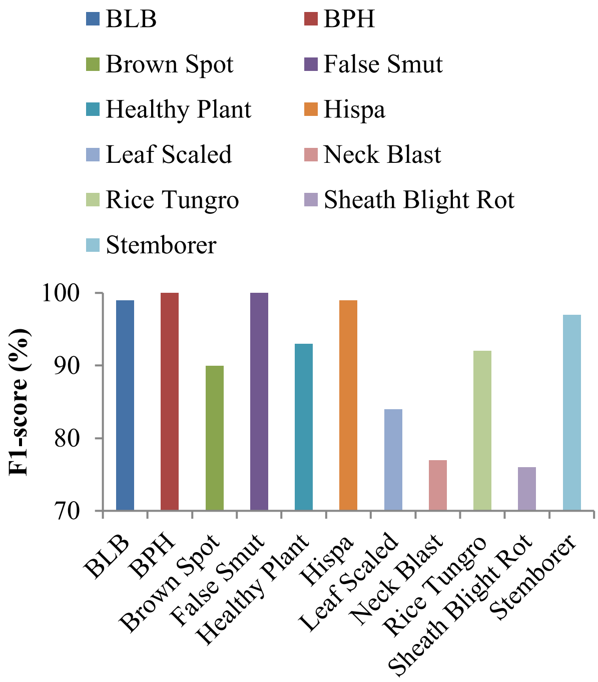
Class	Correctly Classified Instance	Incorrectly Classified Instance	Recall (%)	Precision (%)	F1-score (%)	ROC-AUC score	AP value
BLB	100	2	100	98	99	1.00	1.00
BPH	107	0	99	100	100	1.00	1.00
Brown Spot	113	13	90	90	90	0.99	0.95
False Smut	115	1	100	99	100	1.00	1.00
Healthy Plant	114	7	93	94	93	1.00	0.99
Hispa	113	0	98	100	99	1.00	1.00
Leaf Scaled	92	9	79	91	84	0.99	0.93
Neck Blast	85	32	81	73	77	0.98	0.85
Paddy Tungro	104	14	95	88	92	0.99	0.97
Sheath Blight Rot	75	25	77	75	76	0.98	0.87
Stemborer	107	0	95	100	97	1.00	1.00
Average	102.27	9.36	91.55	91.64	91.55	0.994	0.96



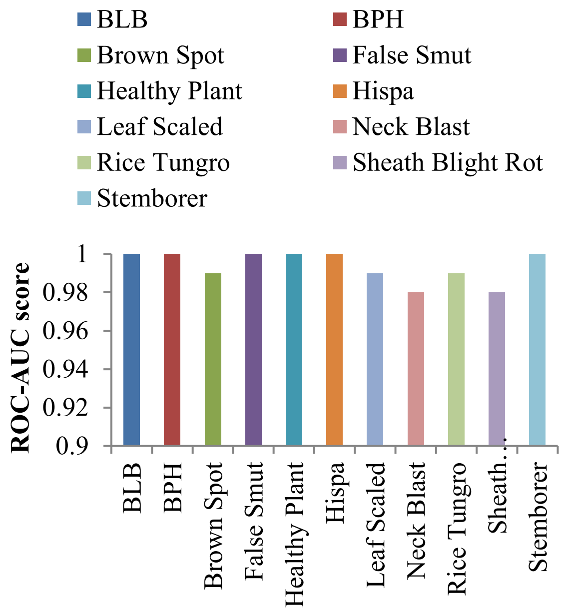


(a) Recall

(b) Precision



(c) F1-score



(d) ROC-AUC score

(e) Average Precision value

Figure 4.15: Performance metrics for each class using CNN_APSO

d) CNN_WWO

The proposed 26-layer CNN model underwent rigorous evaluation using WWO algorithms, resulting in the determination of crucial hyperparameters. Specifically, the hyperparameter values for the 26-layer CNN model using WWO are as follows: a learning rate of 0.002, a batch size of 114, an epoch number of 77, and dropout rates of 0.6 for the CNN model, 0.4 for the attention layers, and 0.5 for the attention_1 layer within this model. Figure 4.16 visually depicts the confusion matrix generated for this model. Within this matrix, correctly predicted instances, highlighted by blue cells, align the model's output with the target values, while white cells represent instances that were misclassified. The model achieved precise classification for 101 instances of BLB, 106 instances of BPH, 108 instances of brown spot, 116 instances of false smut, 116 instances of healthy plants, 111 instances of hispa, 82 instances of leaf scaled, 83 instances of neck blast, 107 instances of tungro, 78 instances of sheath blight rot, and 105 instances of stemborer. However, misclassifications occurred, with 1 instance incorrectly classified as BLB, 1 as BPH, 18 as brown spot, 0 as false smut, 5 as healthy plants, 2 as hispa, 19 as leaf scaled, 34 as neck blast, 11 as tungro, 22 as sheath blight rot, and 2 as stemborer.

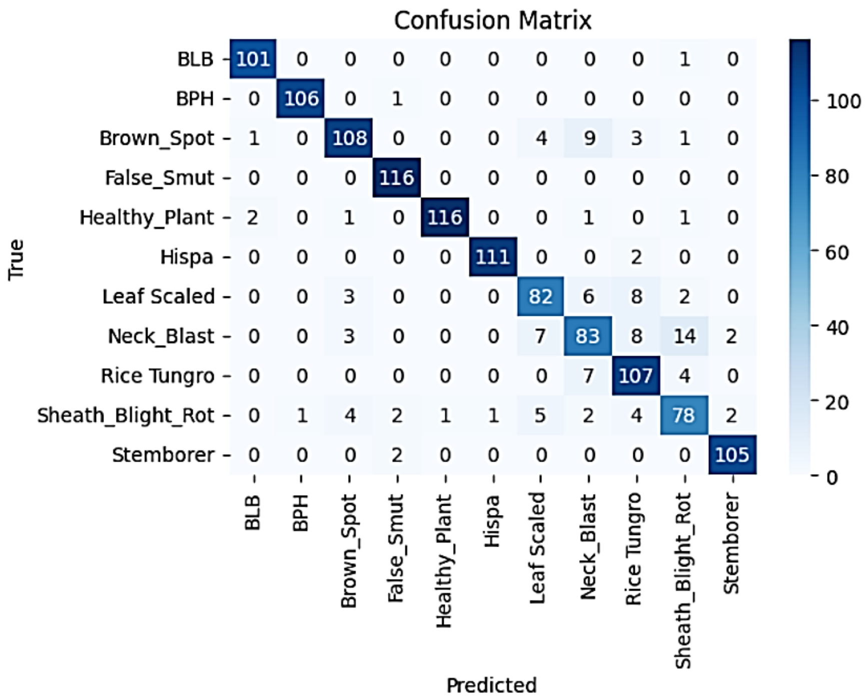


Figure 4.16: Confusion matrix of CNN_WWO

Figure 4.17 presents the ROC curve for CNN_WWO, illustrating the trade-off between recall (True Positive Rate) and the False Positive Rate at various classification thresholds. Each point on this curve represents a specific threshold value. Impressively, the ROC-AUC scores for BLB, BPH, brown spot, false smut, healthy plant, hispa, leaf scaled, neck blast, tungro, sheath blight rot, and stemborer are outstanding, with values of 1.00, 1.00, 0.99, 1.00, 1.00, 1.00, 0.99, 0.98, 0.99, 0.98, and 1.00, respectively. These exceptional scores underscore the model's remarkable ability to accurately detect these specific diseases. BLB, BPH, false smut, healthy plant, hispa, and stemborer achieve the highest ROC-AUC scores, as depicted in Figure 4.20(d), while other classes also demonstrate strong performance.

Figure 4.18 provides insights into the trade-off between precision (the accuracy of positive predictions) and recall (the ability to identify all positive cases) as classification thresholds vary. Notably, the Average Precision (AP) scores for BLB, BPH, brown spot, false smut, healthy plant, hispa, leaf scaled, neck blast, tungro, sheath blight rot, and stemborer are exceptional, with values of 1.00, 1.00, 0.94, 1.00, 0.99, 1.00, 0.92, 0.83, 0.95, 0.88, and 0.99, respectively. These outstanding AP scores are prominently displayed in Figure 18€, highlighting CNN_APSO's superior performance in effectively classifying positive instances while minimizing false positives for BLB, BPH, false smut, and hispa. Conversely, the neck blast class exhibits the lowest AP score, as depicted in Figure 4.20(e).

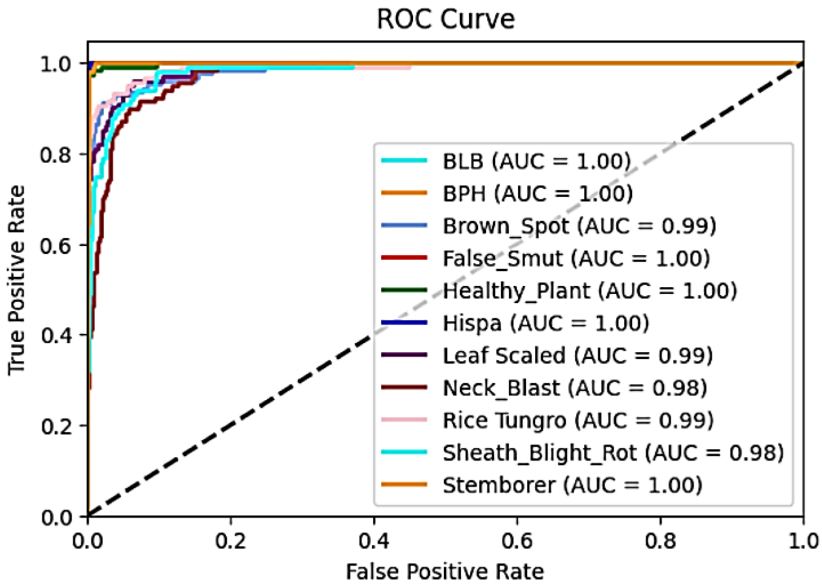


Figure 4.17: ROC curve of CNN_WWO

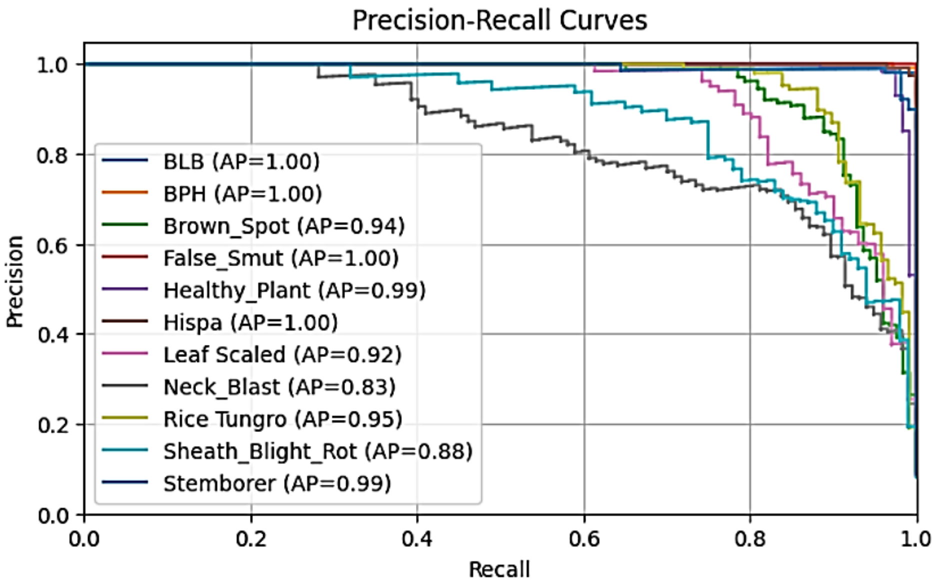


Figure 4.18: Precision-Recall curves of CNN_WWO

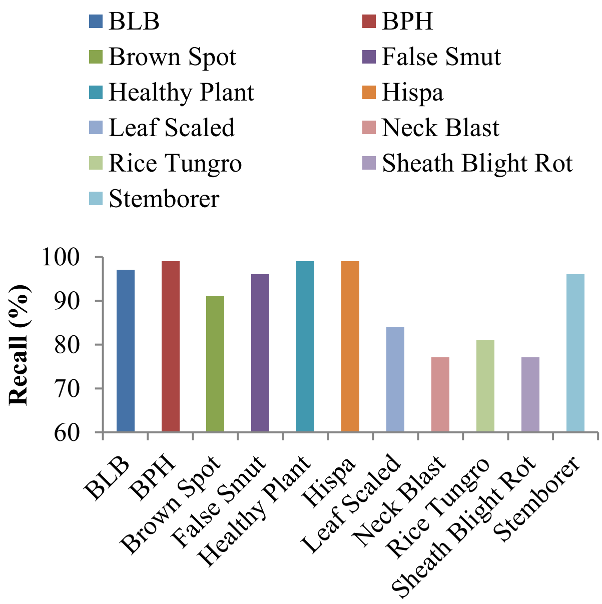
Figure 4.20 provides a visual representation of CNN_WWO's learning curve. During the training process, early stopping was applied, restricting the model to only 13 epochs. The training loss steadily decreased with each epoch, ultimately reaching a final value of 0.0457, indicating effective learning from the training dataset. However, the testing loss displayed fluctuations across the epochs, concluding with a final testing loss of 0.5704. This pattern suggests that while the model performed well on the training data, it faced challenges when generalizing to unseen data, as indicated by the higher testing loss.

Figure 4.19: Learning curve of CNN_WWO

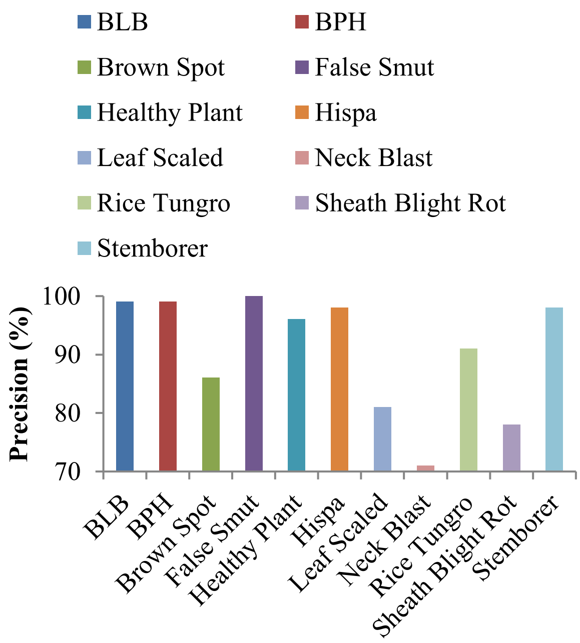
Table 4.4 provides an extensive overview of the classification results achieved by CNN_WWO, presenting various performance metrics for each disease class. In the BLB class, the model attains a recall, precision, F1-score, ROC_AUC score, and AP value of 97%, 99%, 98%, 1.00, and 1.00, respectively. Similarly, for the BPH class, these metrics are recorded at 99%, 99%, 99%, 1.00, and 1.00, respectively. The brown spot class exhibits slightly lower metrics, with a recall of 91%, precision of 86%, F1-score of 88%, ROC_AUC score of 0.99, and AP value of 0.94. In contrast, the false smut class excels with a recall of 96%, precision of 100%, F1-score of 98%, ROC_AUC score of 1.00, and AP value of 1.00. The healthy plant class maintains strong performance, achieving values of 99% recall, 96% precision, 97% F1-score, ROC_AUC score of 1.00, and AP value of 0.99. Similarly, the Hispa class stands out with 99% recall, 98% precision, 99% F1-score, ROC_AUC score of 1.00, and AP value of 1.00. Leaf scaled exhibits a recall of 84%, precision of 81%, F1-score of 82%, ROC_AUC score of 0.99, and AP value of 0.92, while the neck blast class records metrics of 77% recall, 71% precision, 74% F1-score, ROC_AUC score of 0.98, and AP value of 0.83. The tungro class showcases a recall of 81%, precision of 91%, F1-score of 86%, ROC_AUC score of 0.99, and AP value of 0.95, while sheath blight rot class attains a recall of 77%, precision of 78%, F1-score of 78%, ROC_AUC score of 0.98, and AP value of 0.88. Finally, the stemborer class demonstrates strong performance with a recall of 96%, precision of 98%, F1-score of 97%, ROC_AUC score of 1.00, and AP value of 1.00. Figure 4.20 visually illustrates these performance metrics for each class using CNN_WWO, with BPH, healthy plant, and Hispa achieving the highest recall at 99%, while neck blast and sheath blight rot display the lowest recall at 77%. False smut leads in precision, each with a perfect score of 100%, while neck blast trails with the lowest recall at 71%. BPH and Hispa shine with a top F1-score of 99%, while neck blast lags behind with a 77% F1-score for the CNN_WWO model.

Table 4.4: Performance table of CNN_WWO

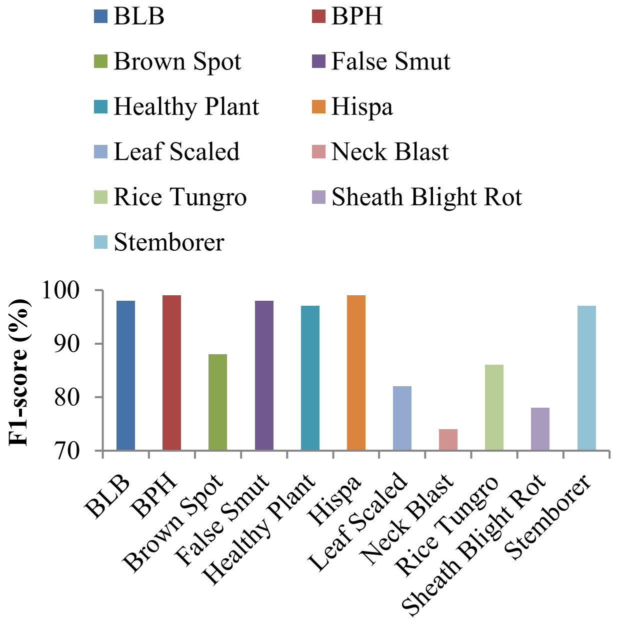
Class	Correctly Classified Instance	Incorrectly Classified Instance	Recall (%)	Precision (%)	F1-score (%)	ROC-AUC score	AP value
BLB	101	1	97	99	98	1.00	1.00
BPH	106	1	99	99	99	1.00	1.00
Brown Spot	108	18	91	86	88	0.99	0.94
False Smut	116	0	96	100	98	1.00	1.00
Healthy Plant	116	5	99	96	97	1.00	0.99
Hispa	111	2	99	98	99	1.00	1.00
Leaf Scaled	82	19	84	81	82	0.99	0.92
Neck Blast	83	34	77	71	74	0.98	0.83
Paddy Tungro	107	11	81	91	86	0.99	0.95
Sheath Blight Rot	78	22	77	78	78	0.98	0.88
Stemborer	105	2	96	98	97	1.00	0.99
Average	101.18	10.45	91.55	90.64	90.55	0.994	0.955



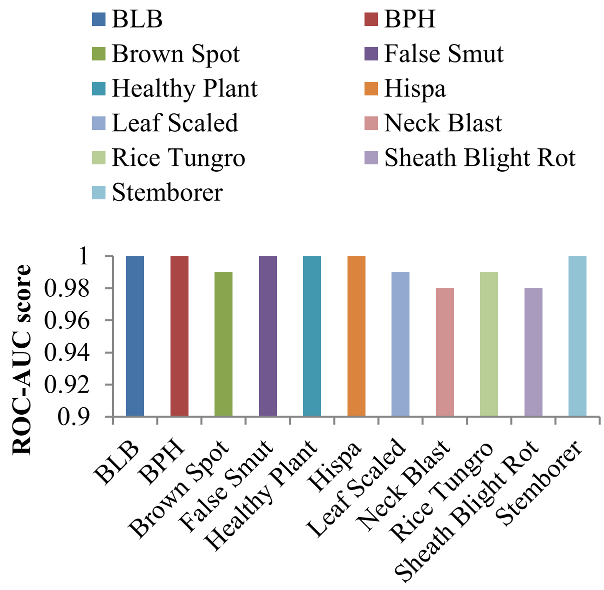
(a) Recall



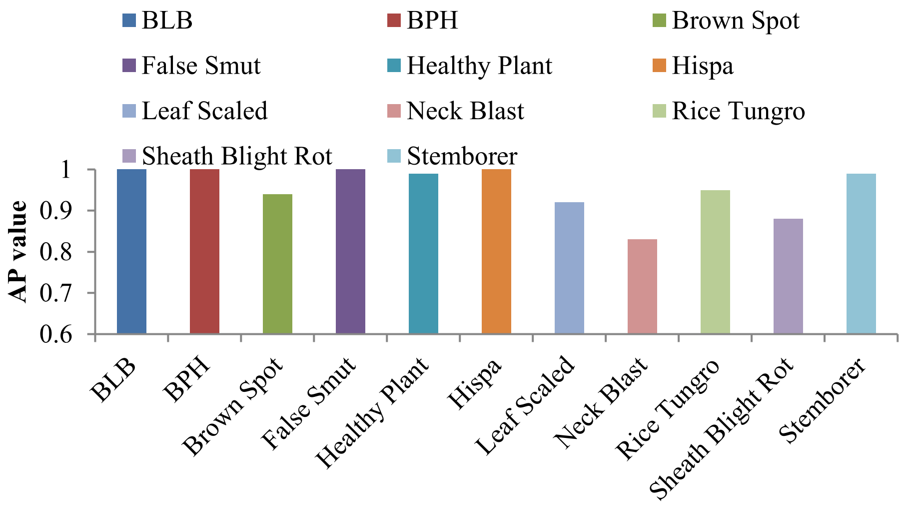
(b) Precision



(c) F1-score



(d) ROC-AUC score



(e) Average Precision value

Figure 4.20: Performance metrics for each class using CNN_WWO

e) CNN_APSWWO

In this section, we provide a comprehensive evaluation of the training outcomes for CNN_APSWWO, offering an in-depth analysis of its performance. The 26-layer CNN model underwent rigorous evaluation using APSWWO algorithms, which led to the determination of crucial hyperparameters. Specifically, the hyperparameter values for the 26-layer CNN model using APSWWO are as follows: a learning rate of 0.001, a batch size of 128, an epoch number of 176, and dropout rates of 0.5 for the CNN model, 0.5 for the attention layers, and 0.4 for the attention_1 layer within this model. Figure 4.21 visually presents the confusion matrix generated for this model. Within this matrix, instances that were correctly predicted are highlighted in blue cells, aligning the model's output with the target values, while white cells represent instances that were misclassified. The model achieved precise classification for 98 instances of BLB, 112 instances of BPH, 97 instances of brown spot, 114 instances of false smut, 111 instances of healthy plants, 109 instances of hispa, 96 instances of leaf scaled, 99 instances of neck blast, 108 instances of tungro, 98 instances of sheath blight rot, and 107 instances of stemborer. However, there were instances of misclassification, with 0 instances incorrectly classified as BLB, 1 as BPH, 6 as brown spot, 2 as false smut, 4 as healthy plants, 0 as hispa, 13 as leaf scaled, 34 as neck blast, 5 as tungro, 14 as sheath blight rot, and 0 as stemborer.

Figure 4.22 displays the ROC curve for CNN_APSWWO, providing insight into the trade-off between recall (True Positive Rate) and the False Positive Rate at various classification thresholds. Each point on this curve corresponds to a specific threshold value. Impressively, the ROC-AUC scores for BLB, BPH, brown spot, false smut, healthy plant, hispa, leaf scaled, neck blast, tungro, sheath blight rot, and stemborer are outstanding, with values of 1.00, 1.00, 1.00, 1.00, 1.00, 1.00, 0.99, 0.98, 1.00, 0.99, and 1.00, respectively. These exceptional scores highlight the model's remarkable ability to accurately detect these specific diseases. BLB, BPH, brown spot, false smut, healthy plant, hispa, tungro, and stemborer achieve the highest ROC-AUC scores, as depicted in Figure 4.25 (d), while other classes also demonstrate strong performance.

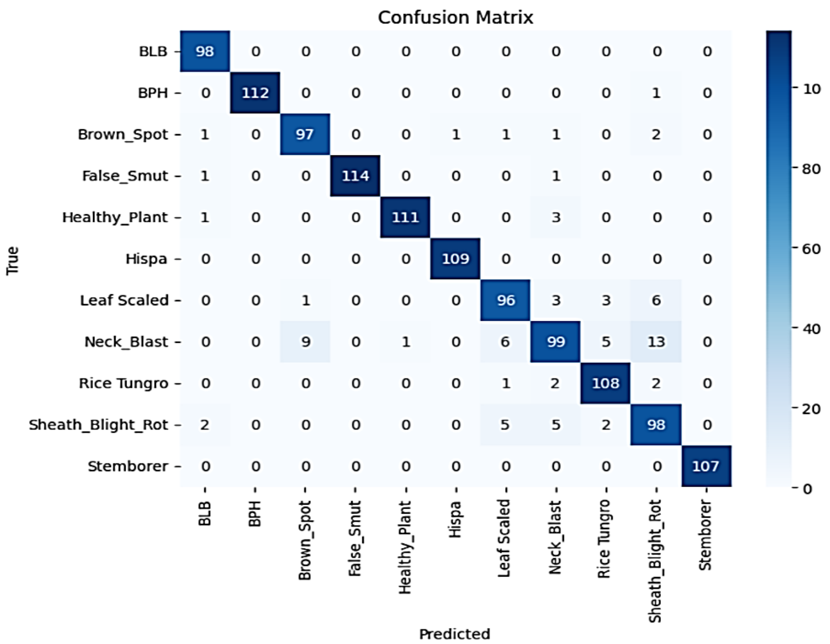


Figure 4.21: Confusion matrix of CNN_APSWWO

Figure 4.23 offers valuable insights into the trade-off between precision (the accuracy of positive predictions) and recall (the ability to identify all positive cases) as classification thresholds vary for CNN_APSWWO. Remarkably, the Average Precision (AP) scores for BLB, BPH, brown spot, false smut, healthy plant, hispa, leaf scaled, neck blast, tungro, sheath blight rot, and stemborer are exceptional, with values of 1.00, 1.00, 1.00, 1.00, 1.00, 1.00, 0.99, 0.98, 1.00, 0.99, and 1.00, respectively. These outstanding AP scores signify

CNN_APSWWO's effective classification of positive instances while minimizing false positives for various disease classes. However, the neck blast and sheath blight rot classes exhibit relatively lower AP scores compared to other classes, as indicated in Figure 4.25 (e).

Figure 4.24 presents the learning curve for CNN_APSWWO. Early stopping was employed, limiting the model to just 9 epochs. Throughout the training, the training loss consistently decreased with each epoch, reaching a final value of 0.0019, indicating effective learning from the training data. However, the testing loss exhibited fluctuations during the epochs, resulting in a final testing loss of 0.4116. This suggests that while the model performed well on the training data, it encountered challenges when generalizing to unseen data, as evidenced by the higher testing loss.

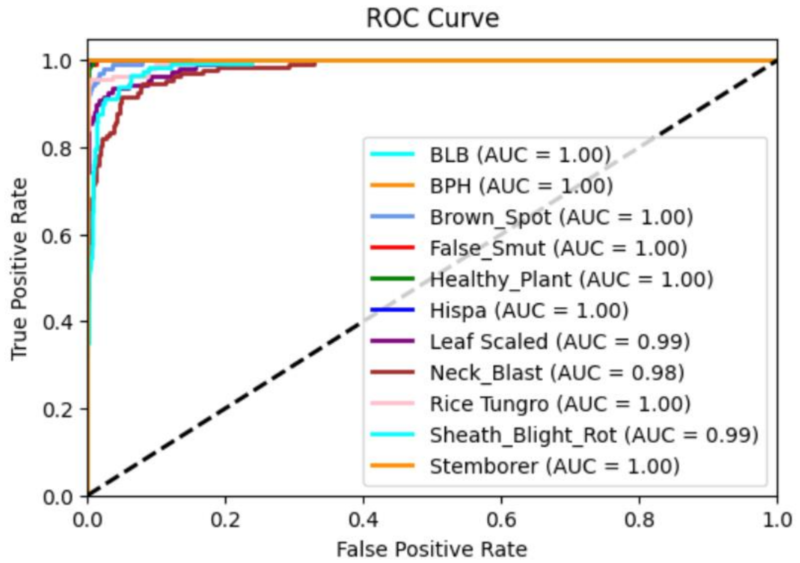


Figure 4.22: ROC curve of CNN_APSWWO

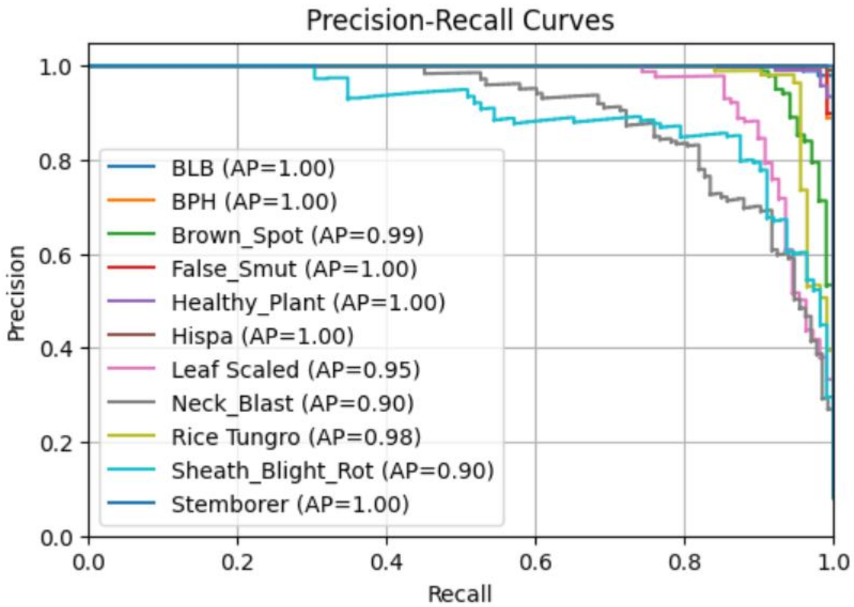


Figure 4.23: Precision-Recall curves of CNN_APSWWO

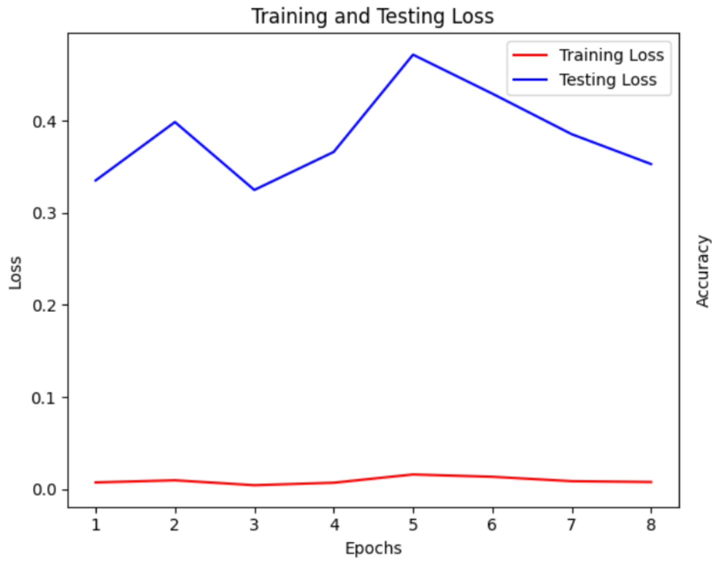
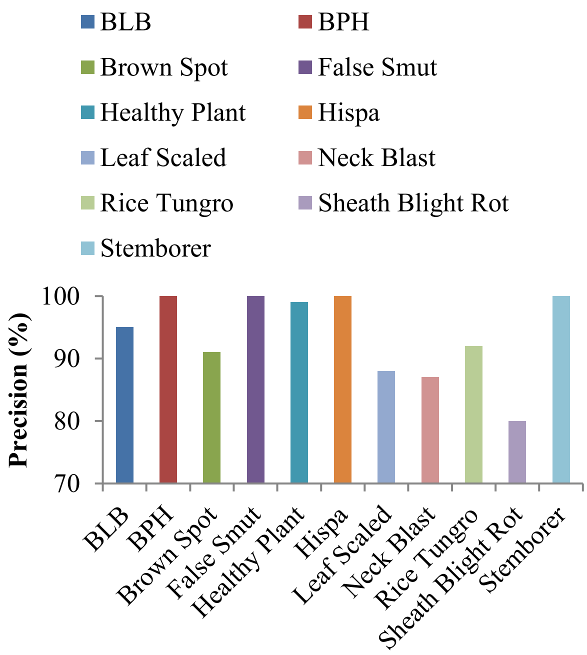


Figure 4.24: Learning curve of CNN_APSWWO

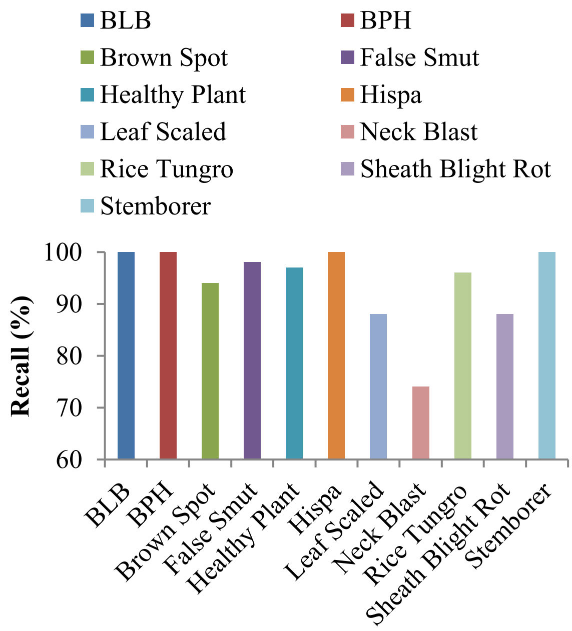
Table 4.5 presents a detailed overview of the classification results achieved by CNN_APSWWO, encompassing various performance metrics for each disease class. In the BLB class, the model achieves a recall, precision, F1-score, ROC_AUC score, and AP value of 95%, 100%, 98%, 1.00, and 1.00, respectively. Similarly, for the BPH class, these metrics are recorded at 100%, 100%, 100%, 1.00, and 1.00, respectively. The brown spot class exhibits slightly lower metrics, with a recall of 91%, precision of 94%, F1-score of 92%, ROC_AUC score of 1.00, and AP value of 0.99. In contrast, the false smut class excels with a recall of 100%, precision of 98%, F1-score of 99%, ROC_AUC score of 1.00, and AP value of 1.00. The healthy plant class maintains strong performance, achieving values of 99% recall, 97% precision, 98% F1-score, ROC_AUC score of 1.00, and AP value of 1.00. Similarly, the Hispa class stands out with 100% recall, 100% precision, 100% F1-score, ROC_AUC score of 1.00, and AP value of 1.00. Leaf scaled exhibits a recall of 88%, precision of 88%, F1-score of 88%, ROC_AUC score of 0.99, and AP value of 0.95, while the neck blast class records metrics of 87% recall, 74% precision, 80% F1-score, ROC_AUC score of 0.98, and AP value of 0.90. The tungro class showcases a recall of 92%, precision of 96%, F1-score of 94%, ROC_AUC score of 1.00, and AP value of 0.98, while sheath blight rot class attains a recall of 80%, precision of 88%, F1-score of 84%, ROC_AUC score of 0.99, and AP value of 0.90. Finally, the stemborer class demonstrates strong performance with a recall of 100%, precision of 100%, F1-score of 100%, ROC_AUC score of 1.00, and AP value of 1.00. Figure 4.25 visually illustrates these performance metrics for each class using CNN_APSWWO, with BPH, false smut, Hispa, and stemborer achieving the highest recall at 100%, while sheath blight rot displays the lowest recall at 80%. BLB, BPH, Hispa, and stemborer lead in precision, each with a perfect score of 100%, while neck blast trails with the lowest precision at 74%. BPH, Hispa, and stemborer shine with a top F1-score of 100%, while neck blast lags behind with an 80% F1-score for the CNN_APSWWO model.

Table 4.5: Performance table of CNN_APSWWO

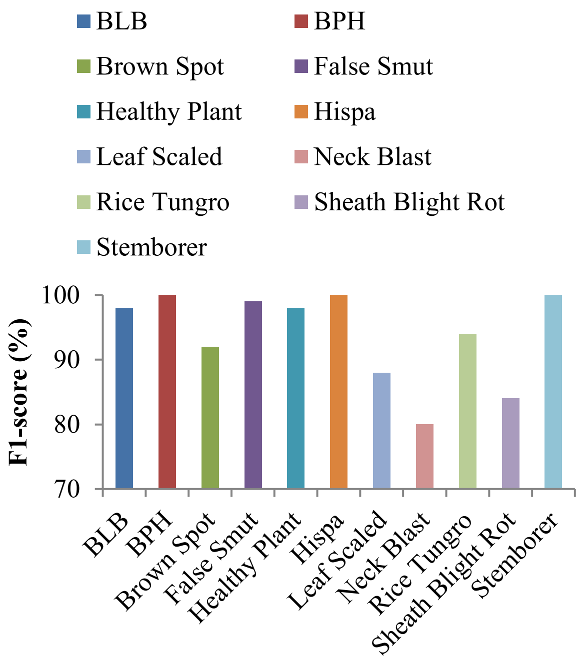
Class	Correctly Classified Instance	Incorrectly Classified Instance	Recall (%)	Precision (%)	F1-score (%)	ROC-AUC score	AP value
BLB	98	0	95	100	98	1.00	1.00
BPH	112	1	100	100	100	1.00	1.00
Brown Spot	97	6	91	94	92	1.00	0.99
False Smut	114	2	100	98	99	1.00	1.00
Healthy Plant	111	4	99	97	98	1.00	1.00
Hispa	109	0	100	100	100	1.00	1.00
Leaf Scaled	96	13	88	88	88	0.99	0.95
Neck Blast	99	34	87	74	80	0.98	0.90
Paddy Tungro	108	5	92	96	94	1.00	0.98
Sheath Blight Rot	98	14	80	88	84	0.99	0.90
Stemborer	107	0	100	100	100	1.00	1.00
Average	104.45	7.18	93.82	94.09	93.91	0.996	0.975



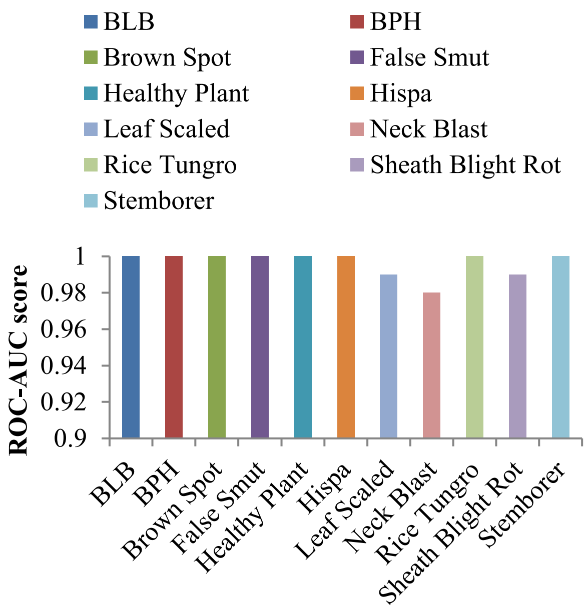
(a) Recall



(b) Precision



(c) F1-score



(d) ROC-AUC score

(e) Average Precision value

Figure 4.25: Performance metrics for each class using CNN_APSWWO

4.2.2 Light Weight Vision Transformer

In this section, the performance analysis and interpretability of LiteViT_Original, LiteViT_PSO, LiteViT_APSO, LiteViT_WWO, and LiteViT_APSWWO models are presented. These variants have been rigorously evaluated to assess their capabilities and suitability for paddy disease classification tasks. The focus is on classification accuracy, training efficiency, and interpretability, providing insights into the strengths and trade-offs of each model. Furthermore, the performance across diverse datasets is explored, highlighting the impact of optimization techniques on LiteViT's performance in the paddy disease classification.

a) LiteViT

In this sub-section, the training and testing outcomes of the proposed 13-layer LiteViT_Original model have been explored, utilizing various performance metrics. An extensive evaluation of LiteViT_Original's training outcomes is presented, offering a profound understanding of its performance. The proposed 13-layer LiteViT model is assessed without applying any optimization algorithms. Critical hyperparameters, including learning rate, batch size, epochs, number of heads, projection dimension, and dropout rates for both the LiteViT model and its attention layers (attention and attention_1), are explicitly configured to specific values: 0.01, 32, 100, 4, 64, 0.4, 0.5, and 0.2, respectively. Figure 4.26 visually represents the confusion matrix generated for this model. In this matrix, correctly predicted instances are indicated by blue cells, signifying alignment between the model's predictions and the actual target values. Conversely, white cells represent instances that were misclassified. The model demonstrated accurate classification of 88 instances of BLB, 112 instances of BPH, 58 instances of brown spot, 110 instances of false smut, 98 instances of healthy plants, 107 instances of hispa, 78 instances of leaf scaled, 49 instances of neck blast, 75 instances of tungro, 52 instances of sheath blight rot, and 96 instances of stemborer. Nevertheless, there were misclassifications, including 10 instances incorrectly labeled as BLB, 1 as BPH, 45 as brown spot, 6 as false smut, 17 as healthy plants, 2 as hispa, 31 as leaf scaled, 84 as neck blast, 38 as tungro, 60 as sheath blight rot, and 11 as stemborer.

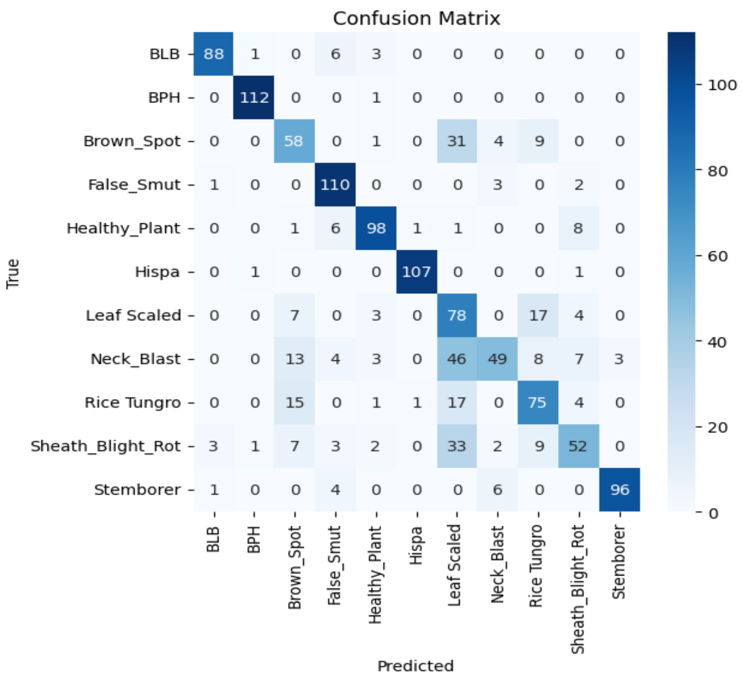


Figure 4.26: Confusion matrix of LiteViT_Original

Figure 4.27 displays the ROC curve of LiteViT_Original, illustrating the trade-off between True Positive Rate (TPR) and False Positive Rate (FPR) across various classification thresholds. Each point on this curve corresponds to a specific threshold value. Remarkably, the ROC-AUC scores for BLB, BPH, brown spot, false smut, healthy plant, hispa, leaf scaled, neck blast, tungro, sheath blight rot, and stemborer are exceptional, with values of 0.99, 1.00, 0.94, 1.00, 0.99, 1.00, 0.92, 0.91, 0.95, 0.90, and 0.99, respectively. This underscores the model's remarkable ability to accurately detect these specific diseases, with BPH, false smut, and hispa achieving the highest ROC-AUC scores, as evident in Figure 4.30 (d). Additionally, other classes also demonstrate robust performance.

Figure 4.28 displays the precision-recall curves for LiteViT_Original in the context of paddy disease classification. These curves offer insights into the trade-off between precision (the accuracy of positive predictions) and recall (the ability to identify all positive cases) as the classification threshold varies. Remarkably, the Average Precision (AP) scores for BLB, BPH, brown spot, false smut, healthy plant, hispa, leaf scaled, neck blast, tungro, sheath blight rot, and stemborer are exceptional, with values of 0.97, 1.00, 0.70, 0.95, 0.93, 1.00, 0.48, 0.65, 0.72, 0.57, and 0.98, respectively. As shown in Figure 4.30(e), BPH and hispa achieve the highest AP scores, while other classes also demonstrate strong performance. Notably, AP values of 1.00 for BPH and hispa indicate LiteViT_Original's effectiveness in accurately classifying positive instances while minimizing false positives in these classes. Conversely, the neck blast class exhibits the lowest AP score, as depicted in Figure 4.30(e).

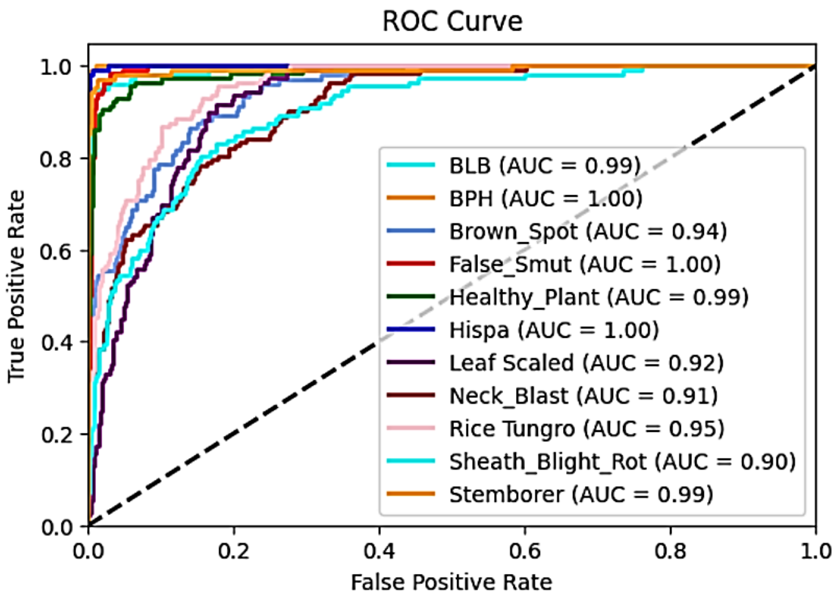


Figure 4.27: ROC curve of LiteViT_Original

Figure 4.29 illustrates the learning curve of LiteViT_Original. The training process employed early stopping, which capped the model's training at 21 epochs. Over the course of this training, the training loss steadily decreased with each epoch, eventually reaching a value of 0.3744. However, the testing loss displayed fluctuations throughout the epochs, ultimately resulting in a final testing loss of 0.7435.

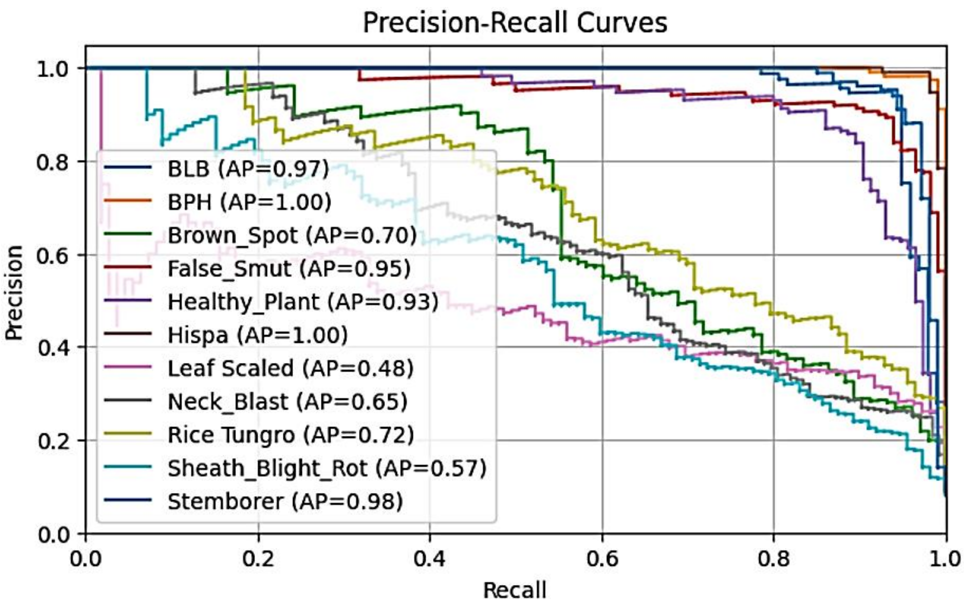


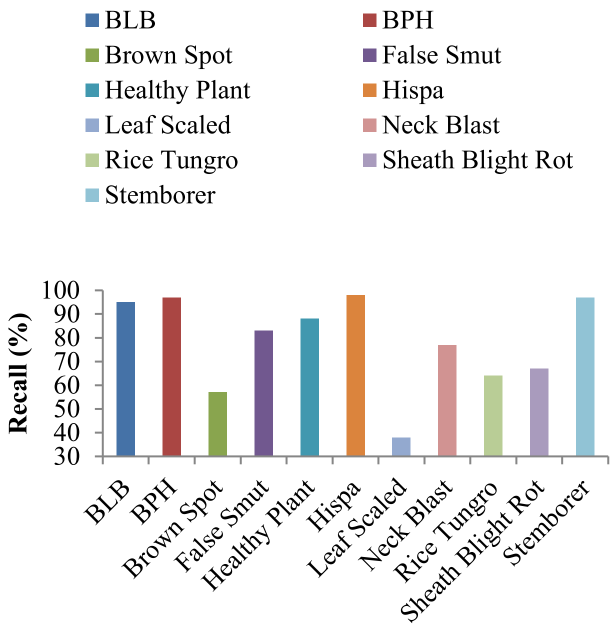
Figure 4.28: Precision-Recall curves of LiteViT_Original

Figure 4.29: Learning curve of LiteViT_Original

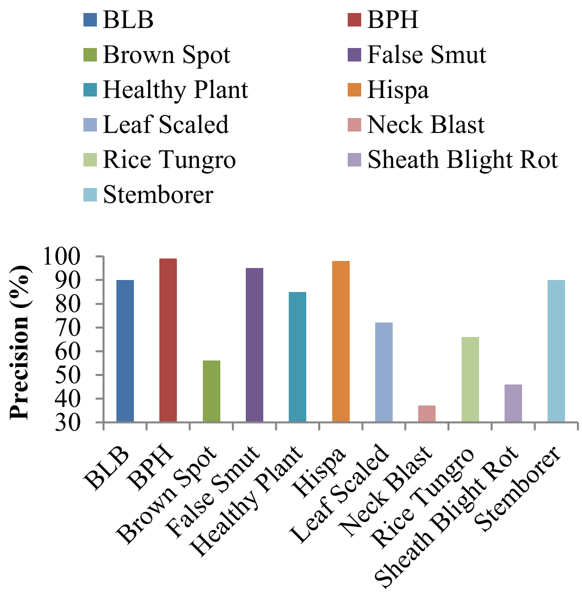
Table 4.6 furnishes an extensive array of classification outcomes for LiteViT_Original, encompassing diverse performance measures for each classification category. Within the BLB category, metrics such as recall, precision, F1-score, ROC_AUC score, and AP value are documented at 95%, 90%, 92%, 0.99, and 0.97, respectively. The BPH category boasts figures of 97%, 99%, 98%, 1.00, and 1.00 for these performance indicators. Meanwhile, the brown spot category presents statistics of 57%, 56%, 57%, 0.94, and 0.70 for recall, precision, F1-score, ROC_AUC score, and AP value correspondingly. In the false smut category, these values climb to 83%, 95%, 88%, 1.00, and 0.95. For the healthy plant category, these performance attributes are captured at 88%, 85%, 86%, 0.99, and 0.93. The Hispa category showcases results of 98%, 98%, 98%, 1.00, and 1.00 across the same metrics. Turning to the leaf scaled category, we find a recall of 38%, precision of 72%, F1-score of 50%, ROC_AUC score of 0.92, and AP value of 0.48. The neck blast category registers a recall of 77%, precision of 37%, F1-score of 50%, ROC_AUC score of 0.91, and AP value of 0.65. The tungro category attains a recall of 64%, precision of 66%, F1-score of 65%, ROC_AUC score of 0.95, and AP value of 0.72, while the sheath blight rot category displays a recall of 67%, precision of 46%, F1-score of 55%, ROC_AUC score of 0.90, and AP value of 0.57. Lastly, the stemborer category showcases a recall of 97%, precision of 90%, F1-score of 93%, ROC_AUC score of 0.99, and AP value of 0.98. Figure 4.30 supplements this data, visually representing the performance metrics for each classification category using LiteViT_Original. Within this visual, Hispa stands out with the highest recall score of 98%, while leaf scaled presents the lowest recall at 38%. In terms of precision, BPH leads the pack with a score of 99%, while neck blast trails behind with the lowest recall at 37%. The top F1-score of 98% is shared by the BPH and Hispa categories, while leaf scaled and neck blast lag behind, both achieving a 50% F1-score.

Table 4.6: Performance table of LiteViT_Original

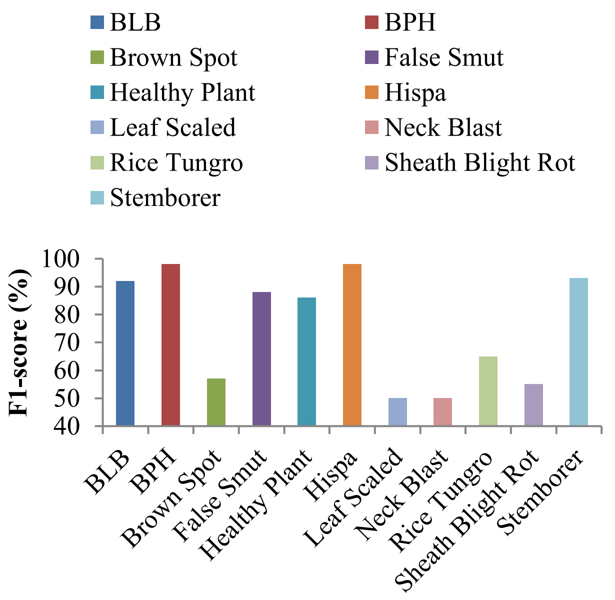
Class	Correctly Classified Instance	Incorrectly Classified Instance	Recall (%)	Precision (%)	F1-score (%)	ROC-AUC score	AP value
BLB	88	10	95	90	92	0.99	0.97
BPH	112	1	97	99	98	1.00	1.00
Brown Spot	58	45	57	56	57	0.94	0.70
False Smut	110	6	83	95	88	1.00	0.95
Healthy Plant	98	17	88	85	86	0.99	0.93
Hispa	107	2	98	98	98	1.00	1.00
Leaf Scaled	78	31	38	72	50	0.92	0.48
Neck Blast	49	84	77	37	50	0.91	0.65
Paddy Tungro	75	38	64	66	65	0.95	0.72
Sheath Blight Rot	52	60	67	46	55	0.90	0.57
Stemborer	96	11	97	90	93	0.99	0.98
Average	83.91	27.73	78.27	75.82	75.64	0.963	0.814



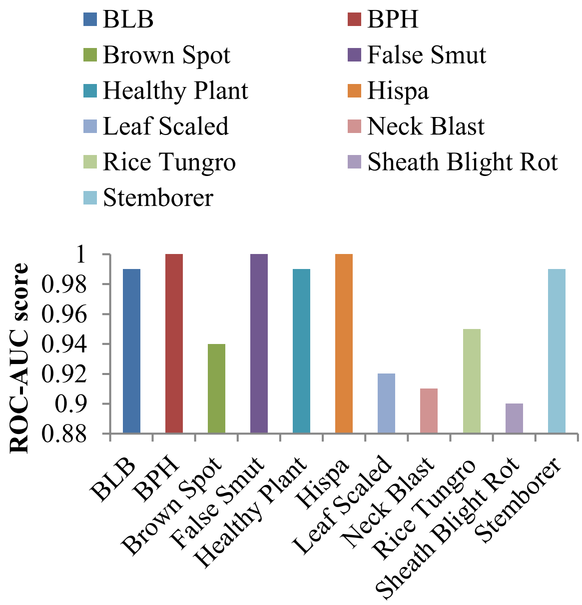
(a) Recall



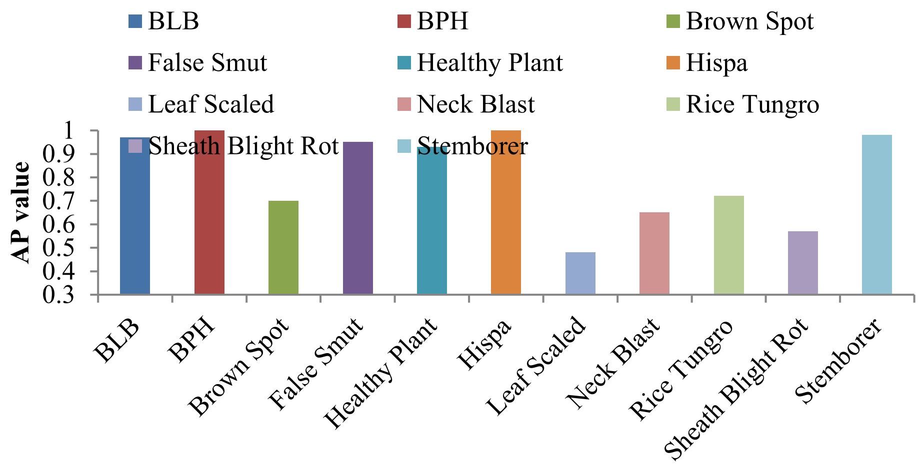
(b) Precision



(c) F1-score



(d) ROC-AUC score



(e) Average Precision value

Figure 4.30: Performance metrics for each class using LiteViT_Original

b) LiteViT_PSO

In this section, the study examines the results of the LiteViT_PSO model, employing a range of performance metrics for evaluation. In this evaluation, the proposed 13-layer LiteViT model was fine-tuned using PSO algorithms, which helped determine essential hyperparameters. Specifically, the hyperparameter configuration for the 13-layer LiteViT model using PSO is as follows: a learning rate of 0.001, a batch size of 166, an epoch count of 122, eight attention heads, a projection dimension of 100, and dropout rates of 0.4 for the LiteViT model itself, 0.2 for the attention layers, and 0.3 for the attention_1 layer within the model. Figure 4.31 illustrates the confusion matrix generated during this evaluation to visualize the model's performance. In this matrix, correctly predicted instances are denoted by blue cells, indicating the model's alignment with the target values. Conversely, white cells represent instances that were misclassified. The model demonstrated precise classifications for various classes: 94 instances of BLB, 109 instances of BPH, 71 instances of brown spot, 115 instances of false smut, 100 instances of healthy plants, 108 instances of hispa, 58 instances of leaf scaled, 48 instances of neck blast, 76 instances of tungro, 61 instances of sheath blight rot, and 105 instances of stemborer. However, there were instances of misclassification as well, including 4 instances incorrectly labeled as BLB, 4 as BPH, 32 as brown spot, 1 as false smut, 15 as healthy plants, 1 as hispa, 51 as leaf scaled, 83 as neck blast, 37 as tungro, 51 as sheath blight rot, and 2 as stemborer.

Figure 4.32 presents the ROC curve for LiteViT_PSO, offering insight into the balance between recall and False Positive Rate at various classification thresholds. Each point on this curve corresponds to a specific threshold value. Notably, LiteViT_PSO exhibits outstanding ROC-AUC scores across multiple classes: BLB, BPH, brown spot, false smut, healthy plant, hispa, leaf scaled, neck blast, tungro, sheath blight rot, and stemborer, achieving values of 1.00, 1.00, 0.96, 1.00, 0.98, 1.00, 0.91, 0.91, 0.95, 0.92, and 1.00, respectively. These remarkable scores emphasize the model's exceptional capability in accurately identifying these specific diseases. Notably, BLB, BPH, false smut, hispa, and stemborer attain the highest ROC-AUC scores, as illustrated in Figure 4.35 (d), while other classes also demonstrate robust performance.

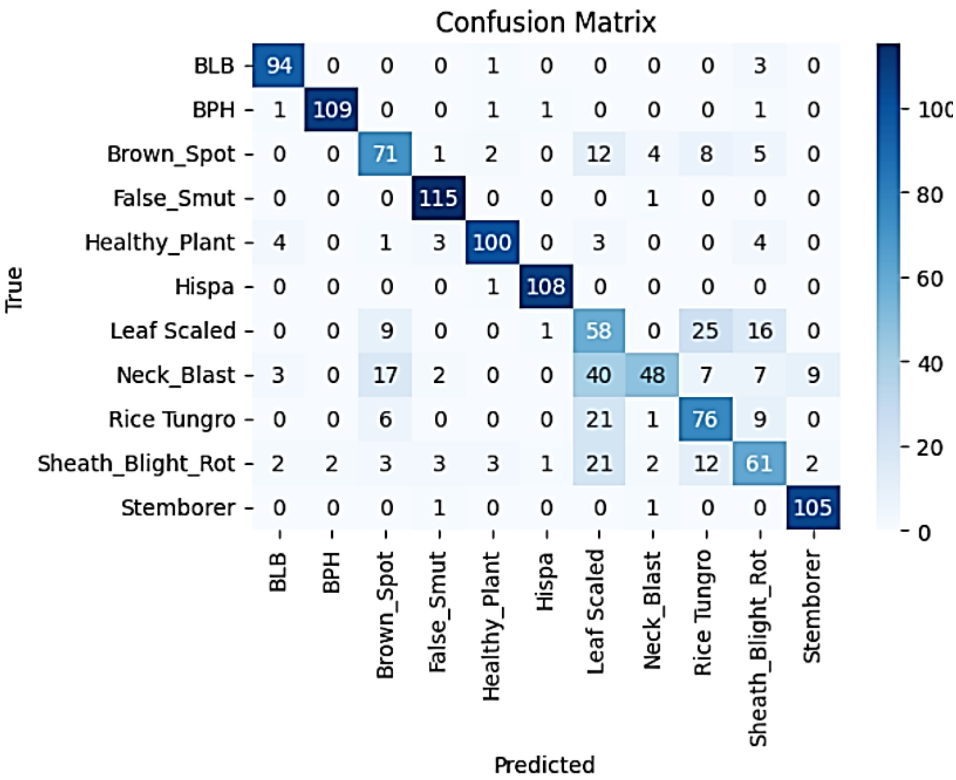


Figure 4.31: Confusion matrix of LiteViT**_PSO**

Figure 4.32 showcases the precision-recall curves for LiteViT_PSO in the context of paddy disease classification, offering valuable insights into the balance between precision (the accuracy of positive predictions) and recall (the ability to identify all positive cases) across various classification thresholds. LiteViT_PSO achieves exceptional Average Precision (AP) scores across multiple classes: BLB, BPH, brown spot, false smut, healthy plant, hispa, leaf scaled, neck blast, tungro, sheath blight rot, and stemborer, with respective values of 0.98, 1.00, 0.78, 0.98, 0.94, 1.00, 0.43, 0.66, 0.68, 0.59, and 0.98. These noteworthy scores are prominently illustrated in Figure 17(e), with BPH and hispa leading in AP scores. These high AP values signify LiteViT_PSO's effectiveness in classifying positive instances while minimizing false positives in these classes. In contrast, the neck blast class exhibits the lowest AP score, as indicated in Figure 4.35 (e).

Figure 4.33 presents the learning curve of LiteViT_PSO, with early stopping limiting the training to only 16 epochs. During this training phase, the training loss steadily decreased with each epoch, ultimately stabilizing at 0.3274. However, in contrast, the testing loss exhibited fluctuating patterns across the epochs, resulting in a final testing loss of 0.7532.

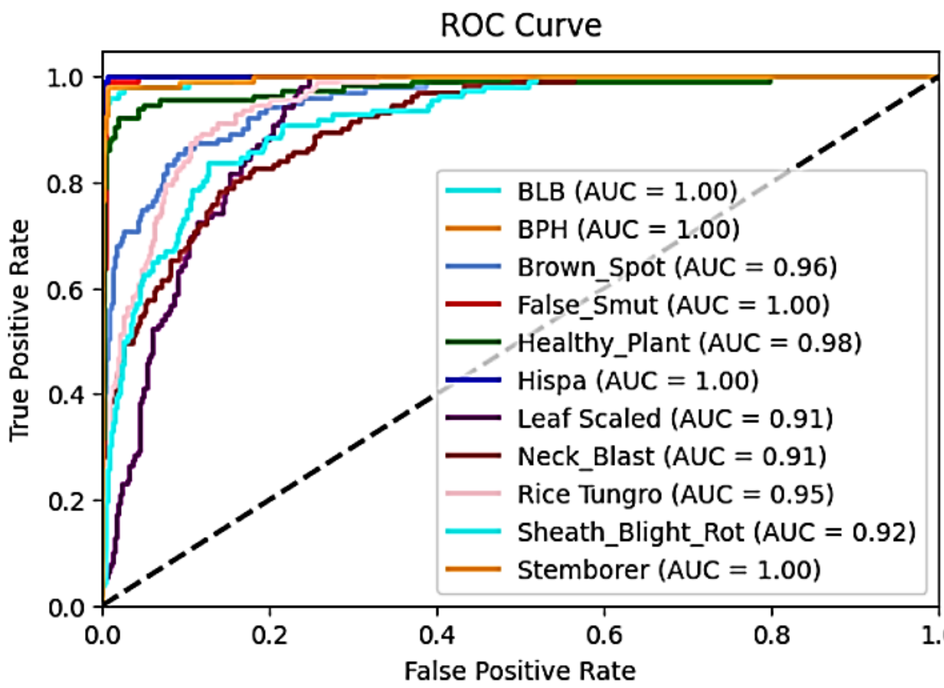


Figure 4.32: ROC curve of LiteViT_PSO

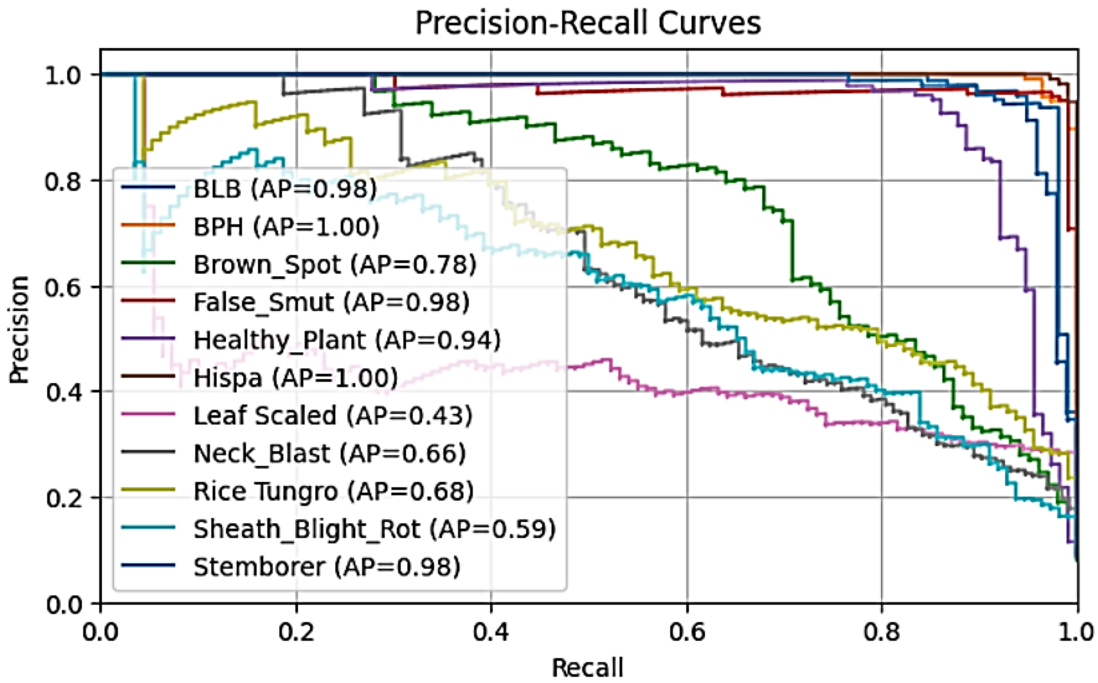


Figure 4.33: Precision-Recall curves of LiteViT_PSO

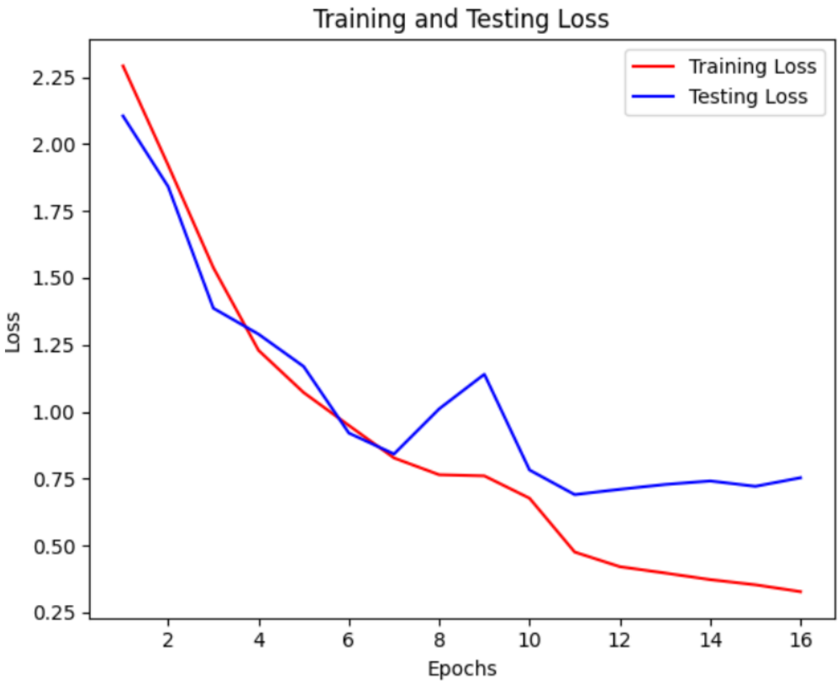
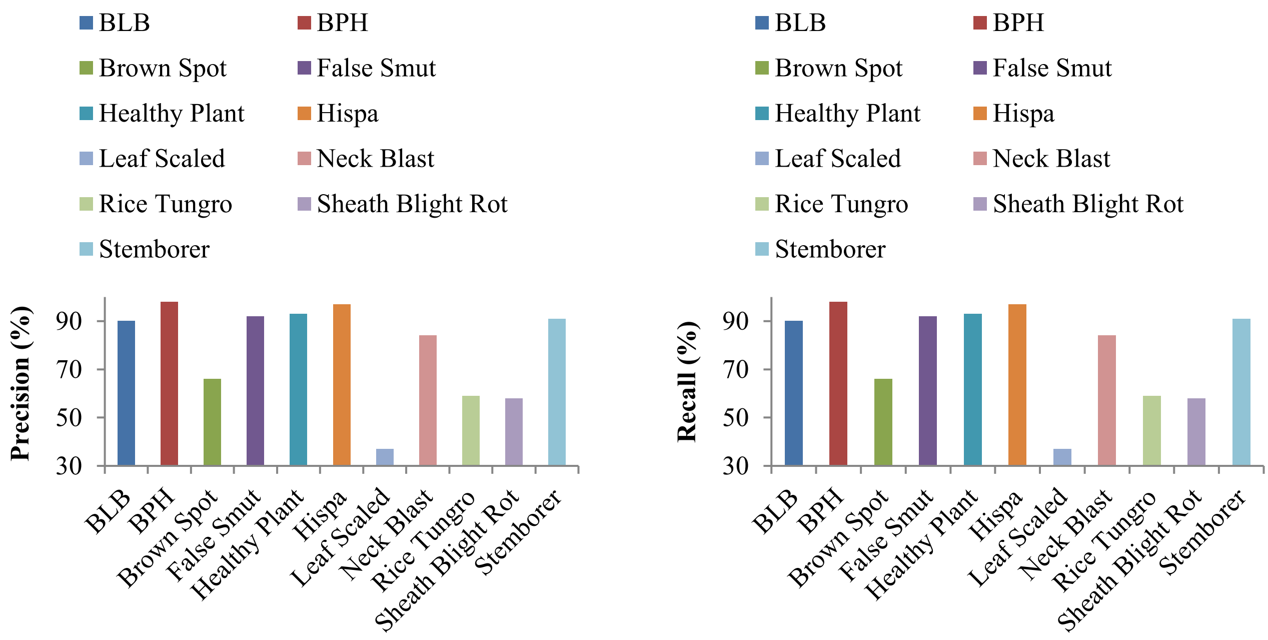


Figure 4.34: Learning curve of LiteViT_PSO

Table 4.7 presents a comprehensive summary of the classification outcomes achieved by LiteViT_PSO, encompassing a range of performance metrics for each disease class. In the case of the BLB class, the model achieves recall, precision, F1-score, ROC_AUC score, and AP value of 90%, 96%, 93%, 1.00, and 0.98, respectively. For the BPH class, the metrics are notably high, with a recall of 98%, precision of 96%, F1-score of 97%, ROC_AUC score of 1.00, and AP value of 1.00. Conversely, the brown spot class exhibits moderate performance, with a recall of 66%, precision of 69%, F1-score of 68%, ROC_AUC score of 0.96, and AP value of 0.78. The false smut class excels, achieving a recall of 92%, precision of 99%, F1-score of 95%, ROC_AUC score of 1.00, and AP value of 0.98. In the healthy plant class, the model maintains strong performance, with values of 93% recall, 87% precision, 90% F1-score, ROC_AUC score of 0.98, and AP value of 0.94. The hispa class stands out with a recall of 97%, precision of 99%, F1-score of 98%, ROC_AUC score of 1.00, and AP value of 1.00. Leaf scaled exhibits a recall of 37%, precision of 53%, F1-score of 44%, ROC_AUC score of 0.91, and AP value of 0.43, while the neck blast class records metrics of 84% recall, 36% precision, 51% F1-score, ROC_AUC score of 0.91, and AP value of 0.66. Tungro class showcases a recall of 59%, precision of 67%, F1-score of 63%, ROC_AUC score of 0.95, and AP value of 0.68, while sheath blight rot class attains a recall of 58%, precision of 54%, F1-score of 56%, ROC_AUC score of 0.92, and AP value of 0.59. Finally, the stemborer class demonstrates robust performance with a recall of 91%, precision of 98%, F1-score of 94%, ROC_AUC score of 1.00, and AP value of 0.98. Figure 4.35 visually illustrates these remarkable performance metrics for each class using LiteViT_PSO, with BPH achieving the highest recall at 98%, and leaf scaled displaying the lowest recall at 37%. Hispa leads in precision, with a score of 99%, while Neck Blast trails with the lowest recall at 36%. Hispa excels with a top F1-score of 98%, while leaf scaled lags behind with a 44% F1-score for the LiteViT_PSO model.

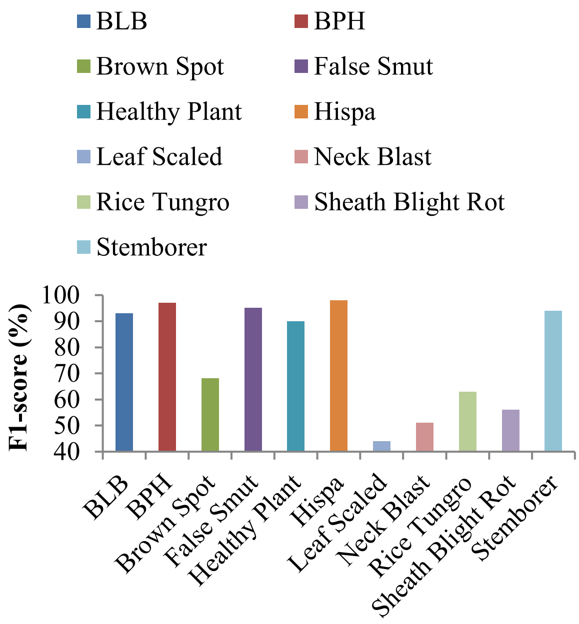
Table 4.7: Performance table of LiteViT_PSO

Class	**Correctly Classified Instance**	**Incorrectly Classified Instance**	**Recall (%)**	**Precision (%)**	**F1-score (%)**	**ROC-AUC score**	**AP value**
BLB	94	4	90	96	93	1.00	0.98
BPH	109	4	98	96	97	1.00	1.00
Brown Spot	71	32	66	69	68	0.96	0.78
False Smut	115	1	92	99	95	1.00	0.98
Healthy Plant	100	15	93	87	90	0.98	0.94
Hispa	108	1	97	99	98	1.00	1.00
Leaf Scaled	58	51	37	53	44	0.91	0.43
Neck Blast	48	83	84	36	51	0.91	0.66
Paddy Tungro	76	37	59	67	63	0.95	0.68
Sheath Blight Rot	61	51	58	54	56	0.92	0.59
Stemborer	105	2	91	98	94	1.00	0.98
Average	85.91	25.55	78.64	77.64	77.18	0.966	0.82

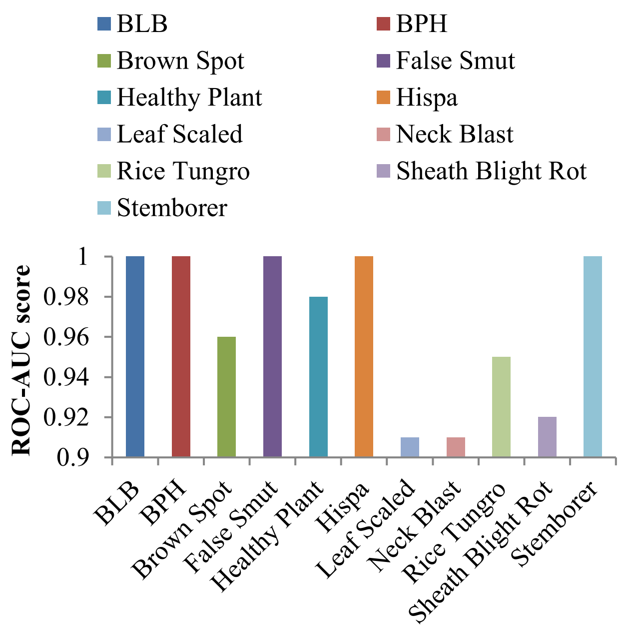


(a) Recall

(b) Precision



(c) F1-score



(d) ROC-AUC score

(e) Average Precision value

Figure 4.35: Performance metrics for each class using LiteViT_PSO

c) LiteViT_APSO

This section offers a comprehensive assessment of the training outcomes for LiteViT_APSO, presenting a detailed analysis of its performance. The 13-layer LiteViT model underwent rigorous evaluation utilizing APSO algorithms, leading to the determination of crucial hyperparameters. Specifically, the hyperparameter configuration for the 18-layer LiteViT model with APSO optimization consists of a learning rate set to 0.001, a batch size of 134, an epoch count of 72, number of heads 64, projection dimension of 252 and dropout rates of 0.4 for the CNN model, 0.2 for the attention layers, and 0.6 for the attention_1 layer within this model. Figure 4.36 visually represents the confusion matrix generated for this model. Within this matrix, correctly predicted instances are denoted by blue cells, aligning the model's output with the target values, while white cells signify instances that were misclassified. The model accurately classified 96 instances of BLB, 112 instances of BPH, 67 instances of brown spot, 112 instances of false smut, 99 instances of healthy plants, 105 instances of hispa, 75 instances of leaf scaled, 61 instances of neck blast, 79 instances of tungro, 72 instances of sheath blight rot, and 103 instances of stemborer. However, there were instances of misclassification, including 2 instances incorrectly classified as BLB, 1 as BPH, 36 as brown spot, 4 as false smut, 16 as healthy plants, 4 as hispa, 34 as leaf scaled, 72 as neck blast, 34 as tungro, 40 as sheath blight rot, and 4 as stemborer.

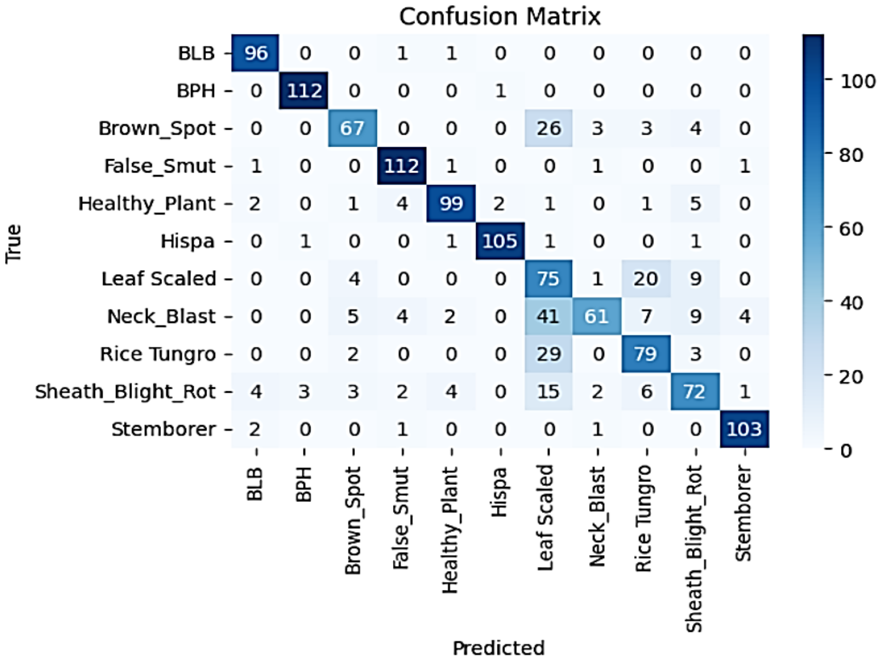


Figure 4.36: Confusion matrix of LiteViT_APSO

Figure 4.37 provides a visual representation of the ROC curve for LiteViT_APSO, which illustrates the trade-off between recall and the False Positive Rate at different classification thresholds. Each point on this curve corresponds to a specific threshold value. Notably, the ROC-AUC scores for BLB, BPH, brown spot, false smut, healthy plant, hispa, leaf scaled, neck blast, tungro, sheath blight rot, and stemborer are outstanding, with values of 1.00, 1.00, 0.98, 1.00, 0.99, 1.00, 0.93, 0.93, 0.97, 0.95, and 1.00, respectively. These exceptional scores underscore the model's remarkable ability to accurately detect these specific diseases. BLB, BPH, false smut, hispa, and stemborer achieve the highest ROC-AUC scores, as depicted in Figure 4.40 (d), while other classes also demonstrate strong performance.

Figure 4.38 offers valuable insights into LiteViT_APSO's precision-recall curves within the context of paddy disease classification. These curves help us understand the balance between precision (the accuracy of positive predictions) and recall (the ability to identify all positive cases) as classification thresholds change. Notably, LiteViT_APSO achieves exceptional Average Precision scores for various classes, including BLB, BPH, brown spot, false smut, healthy plant, hispa, leaf scaled, neck blast, tungro, sheath blight rot, and stemborer, with respective scores of 0.99, 1.00, 0.86, 0.98, 0.96, 0.99, 0.55, 0.71, 0.79, 0.73, and 0.98. These impressive AP scores, particularly for BPH, underscore LiteViT_APSO's effectiveness in accurately classifying positive instances while minimizing false positives in these classes. Conversely, the leaf scaled class exhibits the lowest AP score, as depicted in Figure 4.40 (e).

Figure 4.39 provides a visual representation of LiteViT_APSO's learning curve. Throughout the training process, early stopping was employed, concluding the training after just 16 epochs. The training loss steadily decreased with each epoch, ultimately reaching a final value of 0.2387, indicating effective learning from the training data. However, the testing loss displayed fluctuations across epochs, resulting in a final testing loss of 0.5883. This suggests that while the model demonstrated proficiency on the training data, it may have encountered difficulties in generalizing its performance to unseen data.

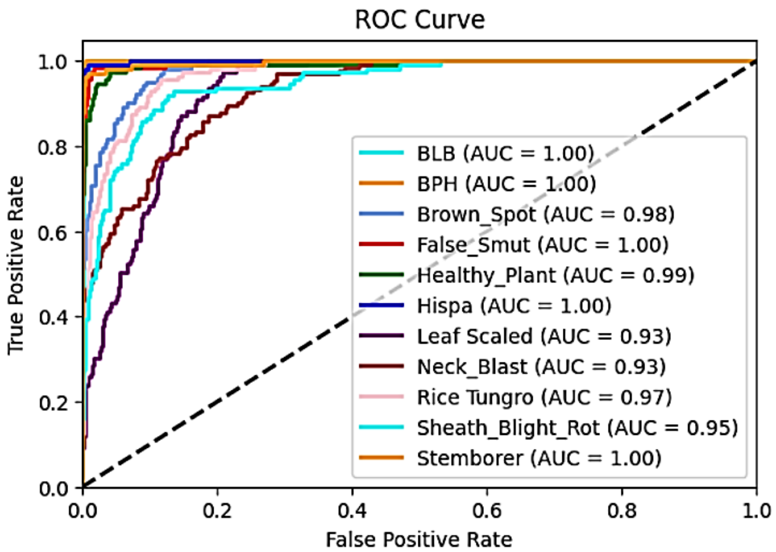


Figure 4.37: ROC curve of LiteViT_APSO

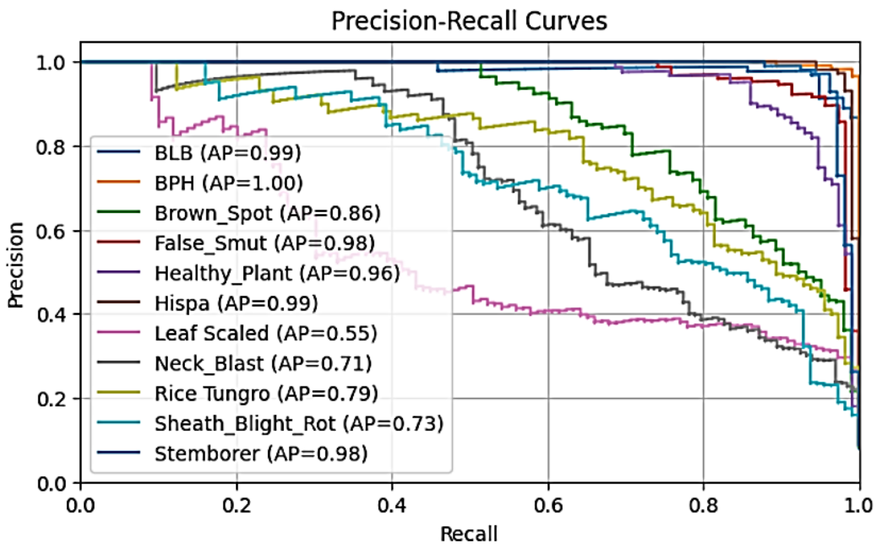


Figure 4.38: Precision-Recall curves of LiteViT_APSO

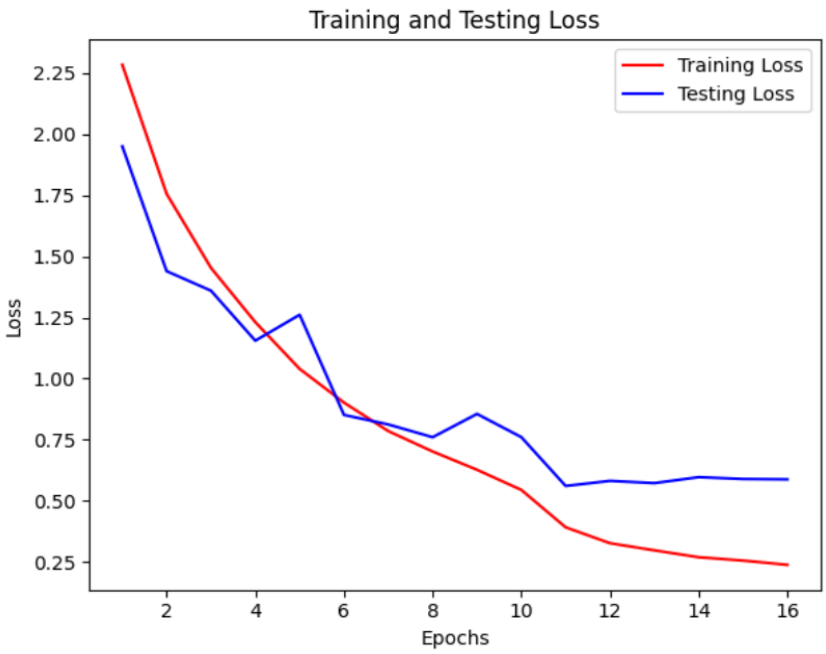
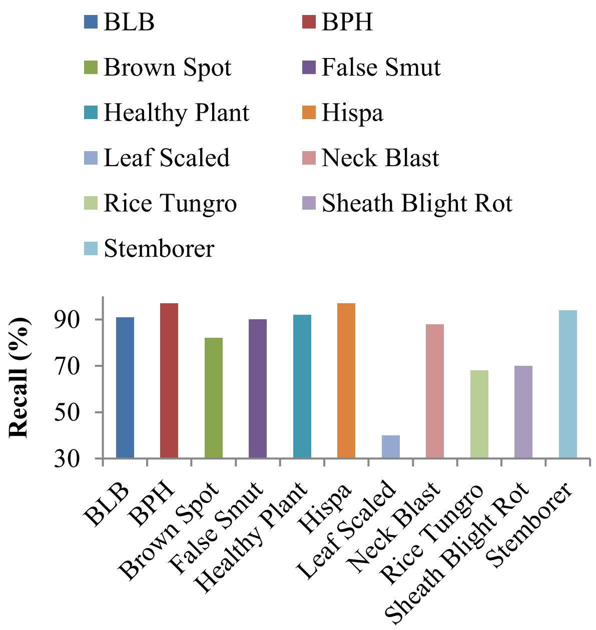


Figure 4.39: Learning curve of LiteViT_APSO

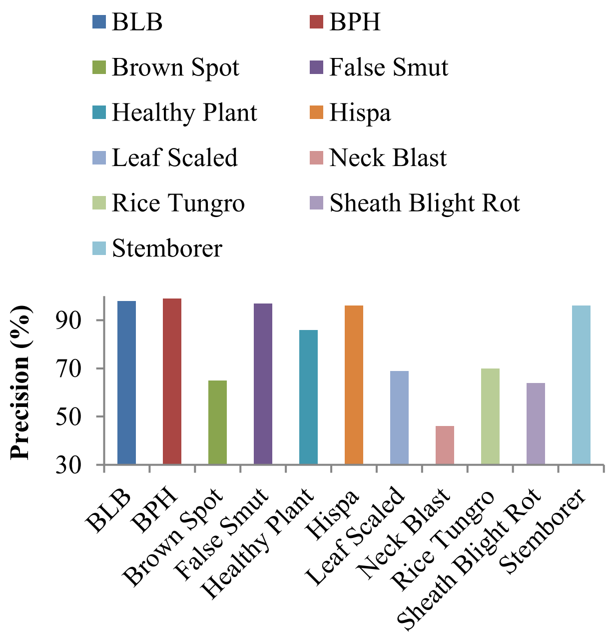
Table 4.8 furnishes a comprehensive summary of the classification outcomes derived from LiteViT_APSO, featuring diverse performance metrics for each disease category. In the BLB category, the model attains a recall, precision, F1-score, ROC_AUC score, and AP value of 91%, 98%, 95%, 1.00, and 0.99, correspondingly. Similarly, for the BPH category, these metrics stand at 97%, 99%, 98%, 1.00, and 1.00, respectively. The brown spot category exhibits somewhat lower metrics with a recall of 82%, precision of 65%, F1-score of 72%, ROC_AUC score of 0.98, and AP value of 0.86. Meanwhile, the false smut category excels with a recall of 90%, precision of 97%, F1-score of 93%, ROC_AUC score of 1.00, and AP value of 0.98. The healthy plant category maintains performance, achieving values of 92% recall, 86% precision, 89% F1-score, ROC_AUC score of 0.99, and AP value of 0.96. Similarly, the Hispa category stands out with 97% recall, 96% precision, 97% F1-score, ROC_AUC score of 1.00, and AP value of 0.99. Leaf scaled exhibits a recall of 40%, precision of 69%, F1-score of 51%, ROC_AUC score of 0.93, and AP value of 0.55, while the neck blast category records metrics of 88% recall, 46% precision, 60% F1-score, ROC_AUC score of 0.93, and AP value of 0.71. The tungro category showcases a recall of 68%, precision of 70%, F1-score of 69%, ROC_AUC score of 0.97, and AP value of 0.79, while sheath blight rot category attains a recall of 70%, precision of 64%, F1-score of 67%, ROC_AUC score of 0.95, and AP value of 0.73. Finally, the stemborer category demonstrates strong performance with a recall of 94%, precision of 96%, F1-score of 95%, ROC_AUC score of 1.00, and AP value of 0.98. Figure 4.40 graphically illustrates these exceptional performance metrics for each category using LiteViT_APSO, with BLB and hispa achieving the highest recall at 97%, while leaf scaled displays the lowest recall at 40%. BPH leads in precision, with a score of 99%, while neck blast trails with the lowest recall at 46%. BPH excels with a top F1-score of 98%, while leaf scaled lags behind with a 51% F1-score for the LiteViT_APSO model.

Table 4.8: Performance table of LiteViT_APSO

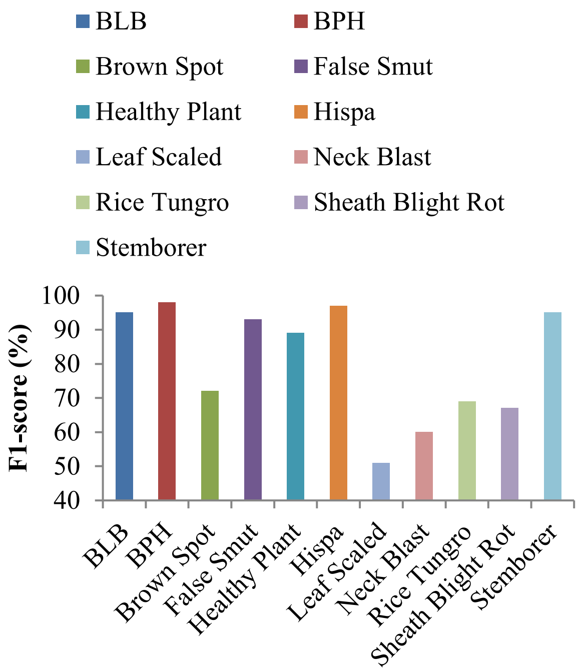
Class	Correctly Classified Instance	Incorrectly Classified Instance	Recall (%)	Precision (%)	F1-score (%)	ROC-AUC score	AP value
BLB	96	2	91	98	95	1.00	0.99
BPH	112	1	97	99	98	1.00	1.00
Brown Spot	67	36	82	65	72	0.98	0.86
False Smut	112	4	90	97	93	1.00	0.98
Healthy Plant	99	16	92	86	89	0.99	0.96
Hispa	105	4	97	96	97	1.00	0.99
Leaf Scaled	75	34	40	69	51	0.93	0.55
Neck Blast	61	72	88	46	60	0.93	0.71
Paddy Tungro	79	34	68	70	69	0.97	0.79
Sheath Blight Rot	72	40	70	64	67	0.95	0.73
Stemborer	103	4	94	96	95	1.00	0.98
Average	89.18	22.45	82.64	80.55	80.55	0.893	0.867



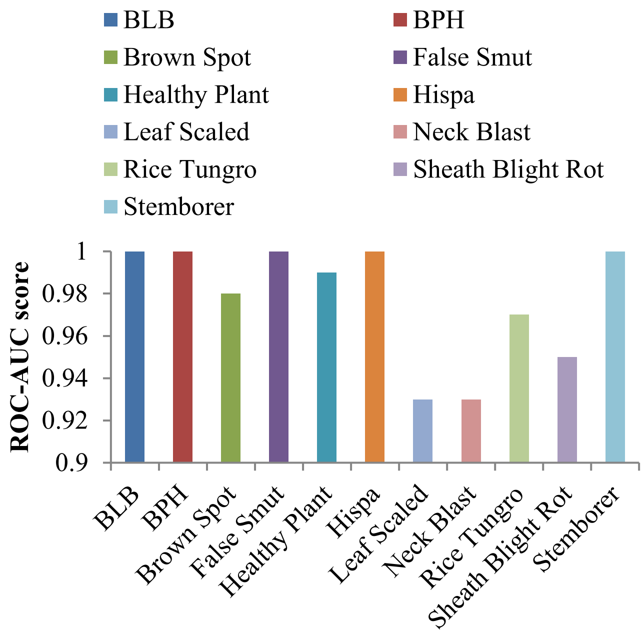
(a) Recall



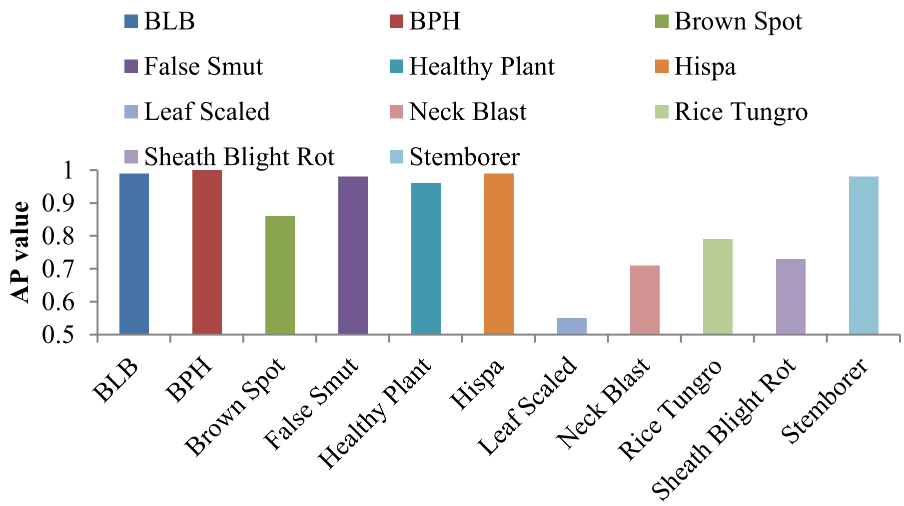
(b) Precision



(c) F1-score



(d) ROC-AUC score



(e) Average Precision value

Figure 4.40: Performance metrics for each class using LiteViT_APSO

d) LiteViT_WWO

The proposed 13-layer LiteViT model underwent rigorous evaluation using WWO algorithms, which led to the determination of crucial hyperparameters. Specifically, the hyperparameter values for the 13-layer LiteViT model using WWO are as follows: a learning rate of 0.004, a batch size of 76, an epoch number of 67, a number of heads of 16, a projection dimension of 95, and dropout rates of 0.5 for the LiteViT model, 0.5 for the attention layers, and 0.2 for the attention_1 layer within this model. Figure 4.41 provides a visual representation of the confusion matrix generated for this model. Within this matrix, correctly predicted instances, highlighted by blue cells, align the model's output with the target values, while white cells represent instances that were misclassified. The model achieved precise classification for 94 instances of BLB, 109 instances of BPH, 71 instances of brown spot, 115 instances of false smut, 100 instances of healthy plants, 108 instances of hispa, 58 instances of leaf scaled, 48 instances of neck blast, 76 instances of tungro, 61 instances of sheath blight rot, and 105 instances of stemborer. However, misclassifications occurred, with 4 instances incorrectly classified as BLB, 4 as BPH, 32 as brown spot, 1 as false smut, 15 as healthy plants, 1 as hispa, 51 as leaf scaled, 85 as neck blast, 37 as tungro, 51 as sheath blight rot, and 2 as stemborer.

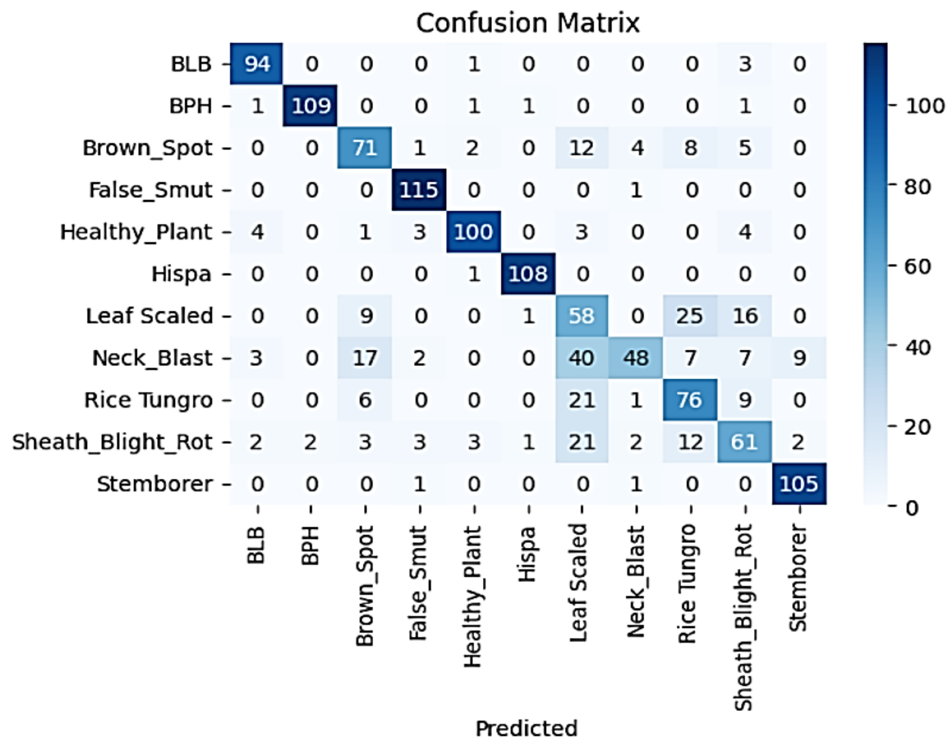


Figure 4.41: Confusion matrix of LiteViT_WWO

Figure 4.42 displays the ROC curve for LiteViT_WWO, portraying the trade-off between recall (True Positive Rate) and the False Positive Rate at various classification thresholds. Each point on this curve corresponds to a specific threshold value. Remarkably, the ROC-AUC scores for BLB, BPH, brown spot, false smut, healthy plant, hispa, leaf scaled, neck blast, tungro, sheath blight rot, and stemborer are exceptional, with values of 1.00, 1.00, 0.96, 1.00, 0.98, 1.00, 0.91, 0.91, 0.95, 0.92, and 1.00, respectively. These outstanding scores underscore the model's remarkable ability to accurately detect these specific diseases. BLB, BPH, false smut, hispa, and stemborer achieve the highest ROC-AUC scores, as depicted in Figure 4.45(d), while other classes also demonstrate strong performance.

Figure 4.43 offers insights into the trade-off between precision (the accuracy of positive predictions) and recall (the ability to identify all positive cases) as classification thresholds vary. Remarkably, the Average Precision (AP) scores for BLB, BPH, brown spot, false smut, healthy plant, hispa, leaf scaled, neck blast, tungro, sheath blight rot, and stemborer are exceptional, with values of 0.98, 1.00, 0.78, 0.98, 0.94, 1.00, 0.43, 0.66, 0.68, 0.59, and 0.98, respectively. These exceptional AP scores are prominently depicted in Figure 18€, underscoring LiteViT_APSO's superior performance in effectively classifying positive instances while minimizing false positives for BPH and hispa. Conversely, the leaf scaled class exhibits the lowest AP score, as indicated in Figure 4.45 (e).

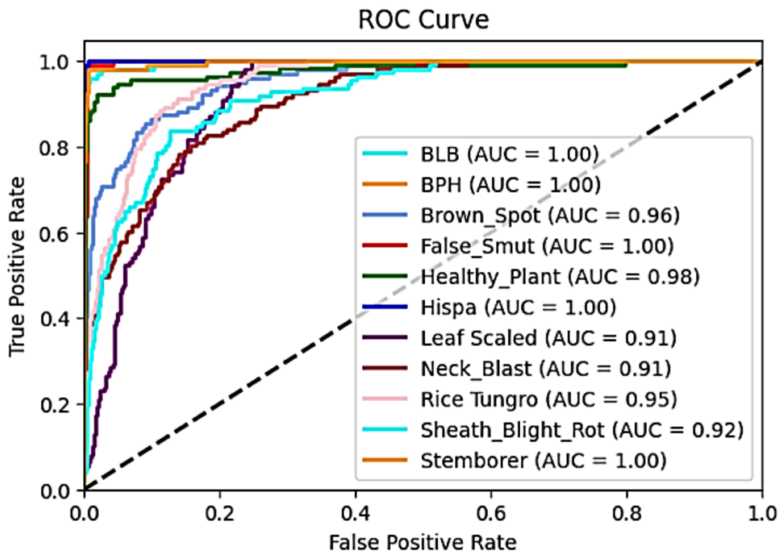


Figure 4.42: ROC curve of LiteViT_WWO

Figure 4.44 visually illustrates LiteViT_WWO's learning curve. Early stopping was employed during the training process, limiting the model to only 16 epochs. The training loss consistently decreased with each epoch, reaching a final value of 0.3274, indicating effective learning from the training dataset. However, the testing loss exhibited fluctuations throughout the epochs, resulting in a final testing loss of 0.7532. This pattern suggests that while the model performed well on the training data, it encountered challenges when generalizing to unseen data, as indicated by the higher testing loss.

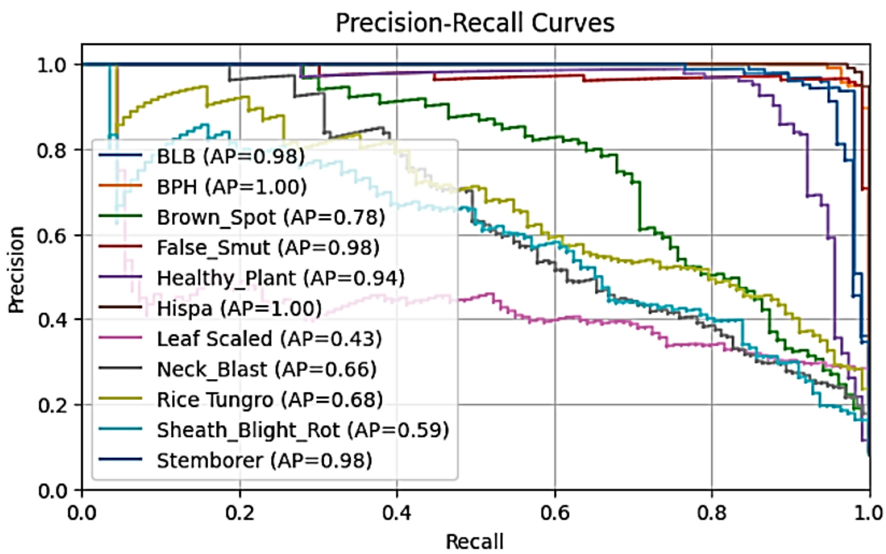


Figure 4.43: Precision-Recall curves of LiteViT_WWO

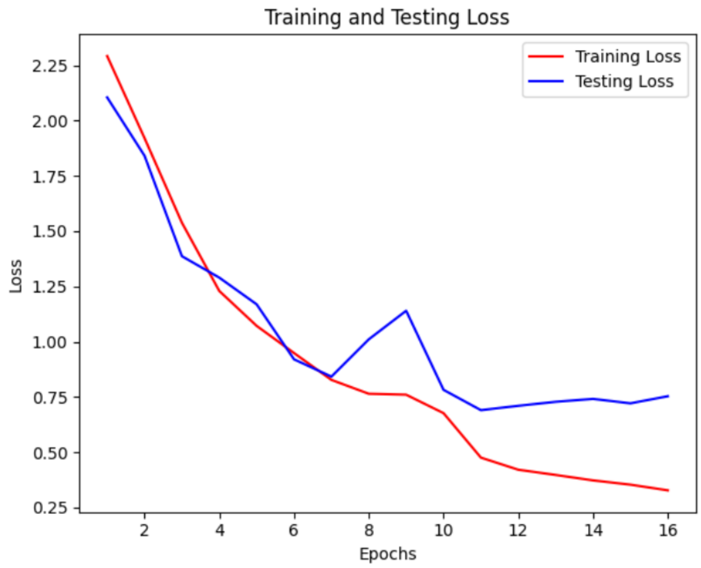
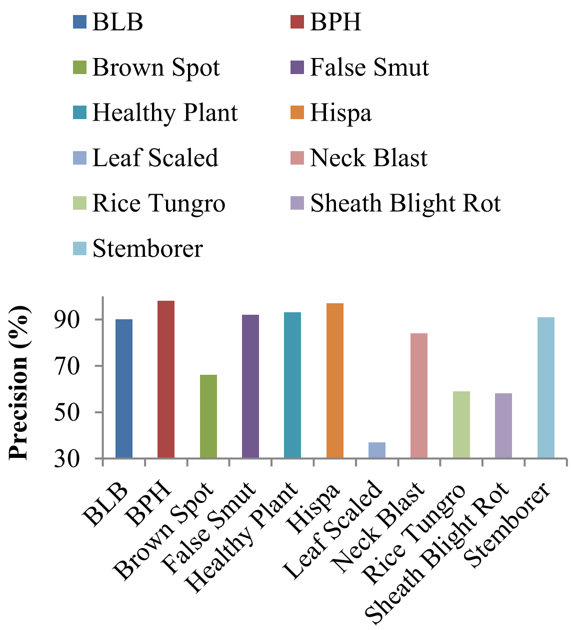


Figure 4.44: Learning curve of LiteViT_WWO

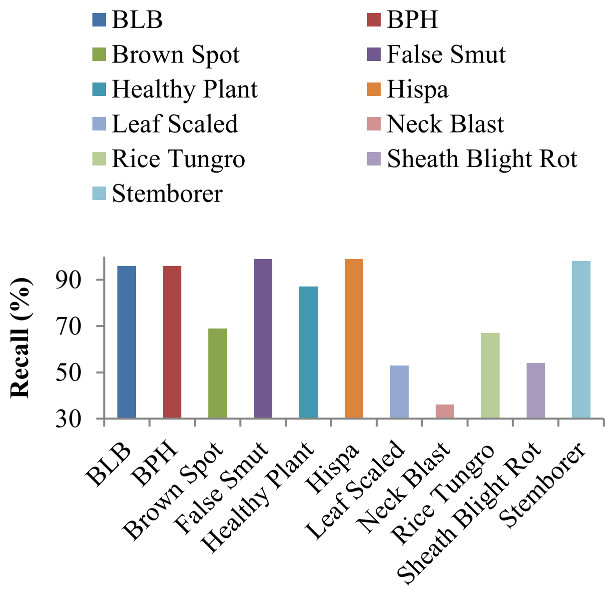
Table 4.9 provides an extensive overview of the classification results achieved by LiteViT_WWO, presenting various performance metrics for each disease class. In the BLB class, the model attains a recall, precision, F1-score, ROC_AUC score, and AP value of 90%, 96%, 93%, 1.00, and 0.98, respectively. Similarly, for the BPH class, these metrics are recorded at 98%, 96%, 97%, 1.00, and 1.00, respectively. The brown spot class exhibits lower metrics, with a recall of 66%, precision of 69%, F1-score of 68%, ROC_AUC score of 0.96, and AP value of 0.78. In contrast, the false smut class excels with a recall of 92%, precision of 99%, F1-score of 95%, ROC_AUC score of 1.00, and AP value of 0.98. The healthy plant class maintains performance, achieving values of 93% recall, 87% precision, 90% F1-score, ROC_AUC score of 0.98, and AP value of 0.94. Similarly, the Hispa class stands out with 97% recall, 99% precision, 98% F1-score, ROC_AUC score of 1.00, and AP value of 1.00. Leaf scaled exhibits a recall of 37%, precision of 53%, F1-score of 44%, ROC_AUC score of 0.91, and AP value of 0.43, while the neck blast class records metrics of 84% recall, 36% precision, 51% F1-score, ROC_AUC score of 0.91, and AP value of 0.66. The tungro class showcases a recall of 59%, precision of 67%, F1-score of 63%, ROC_AUC score of 0.95, and AP value of 0.68, while sheath blight rot class attains a recall of 58%, precision of 54%, F1-score of 56%, ROC_AUC score of 0.92, and AP value of 0.59. Finally, the stemborer class demonstrates strong performance with a recall of 91%, precision of 98%, F1-score of 94%, ROC_AUC score of 1.00, and AP value of 0.98. Figure 4.45 visually illustrates these performance metrics for each class using LiteViT_WWO, with BPH achieving the highest recall at 98%, while leaf scaled display the lowest recall at 37%. False smut and hispa leads in precision, each with a score of 99%, while neck blast trails with the lowest recall at 36%. Hispa shine with a top F1-score of 98%, while leaf scaled lags behind with a 44% F1-score for the LiteViT_WWO model.

Table 4.9: Performance table of LiteViT_WWO

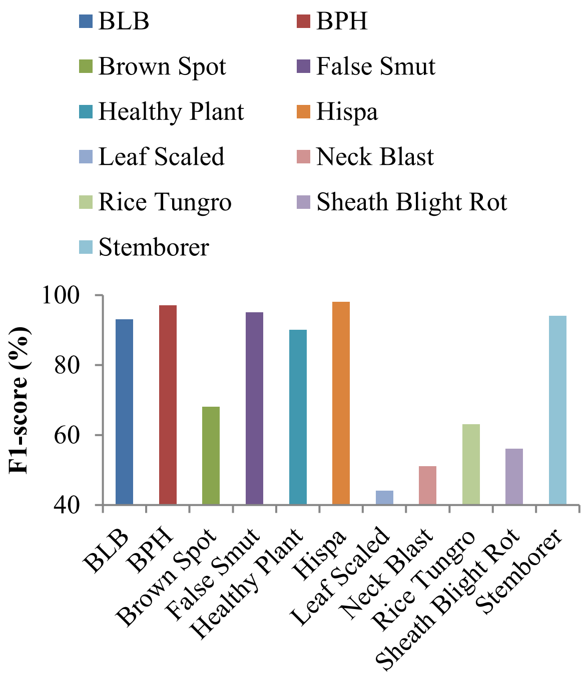
Class	Correctly Classified Instance	Incorrectly Classified Instance	Recall (%)	Precision (%)	F1-score (%)	ROC-AUC score	AP value
BLB	94	4	90	96	93	1.00	0.98
BPH	109	4	98	96	97	1.00	1.00
Brown Spot	71	32	66	69	68	0.96	0.78
False Smut	115	1	92	99	95	1.00	0.98
Healthy Plant	100	15	93	87	90	0.98	0.94
Hispa	108	1	97	99	98	1.00	1.00
Leaf Scaled	58	51	37	53	44	0.91	0.43
Neck Blast	48	85	84	36	51	0.91	0.66
Paddy Tungro	76	37	59	67	63	0.95	0.68
Sheath Blight Rot	61	51	58	54	56	0.92	0.59
Stemborer	105	2	91	98	94	1.00	0.98
Average	85.91	25.73	78.64	77.64	77.18	0.966	0.82



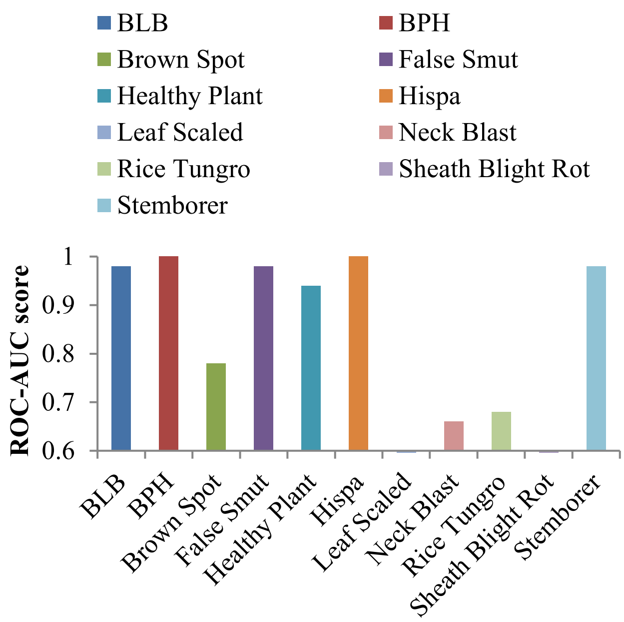
(a) Recall



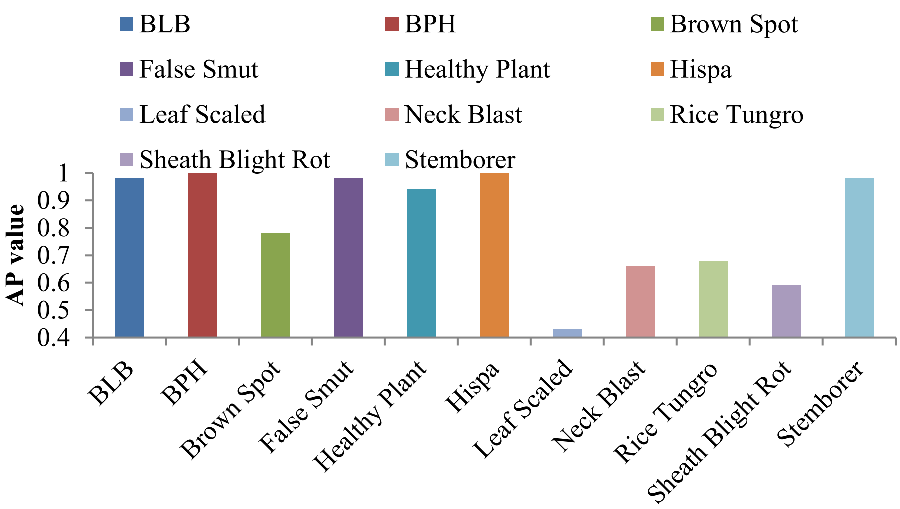
(b) Precision



(c) F1-score



(d) ROC-AUC score



(e) Average Precision value

Figure 4.45: Performance metrics for each class using LiteViT_WWO

e) LiteViT_APSWWO

This section conducts a comprehensive evaluation of the training outcomes for LiteViT_APSWWO, offering an in-depth analysis of its performance. The 13-layer LiteViT model underwent rigorous evaluation using APSWWO algorithms, which led to the determination of crucial hyperparameters. Specifically, the hyperparameter values for the 18-layer LiteViT model using APSWWO are as follows: a learning rate of 0.001, a batch size of 256, an epoch number of 287, number of heads 77, projection dimension of 512, and dropout rates of 0.5 for the LiteViT model, 0.2 for the attention layers, and 0.4 for the attention_1 layer within this model. Figure 4.46 visually presents the confusion matrix generated for this model. Within this matrix, instances that were correctly predicted are highlighted in blue cells, aligning the model's output with the target values, while white cells represent instances that were misclassified. The model achieved precise classification for 96 instances of BLB, 111 instances of BPH, 90 instances of brown spot, 113 instances of false smut, 105 instances of healthy plants, 108 instances of hispa, 77 instances of leaf scaled, 77 instances of neck blast, 103 instances of tungro, 76 instances of sheath blight rot, and 106 instances of stemborer. However, there were instances of misclassification, with 2 instances incorrectly classified as BLB, 2 as BPH, 13 as brown spot, 3 as false smut, 10 as healthy plants, 1 as hispa, 32 as leaf scaled, 56 as neck blast, 10 as tungro, 36 as sheath blight rot, and 1 as stemborer.

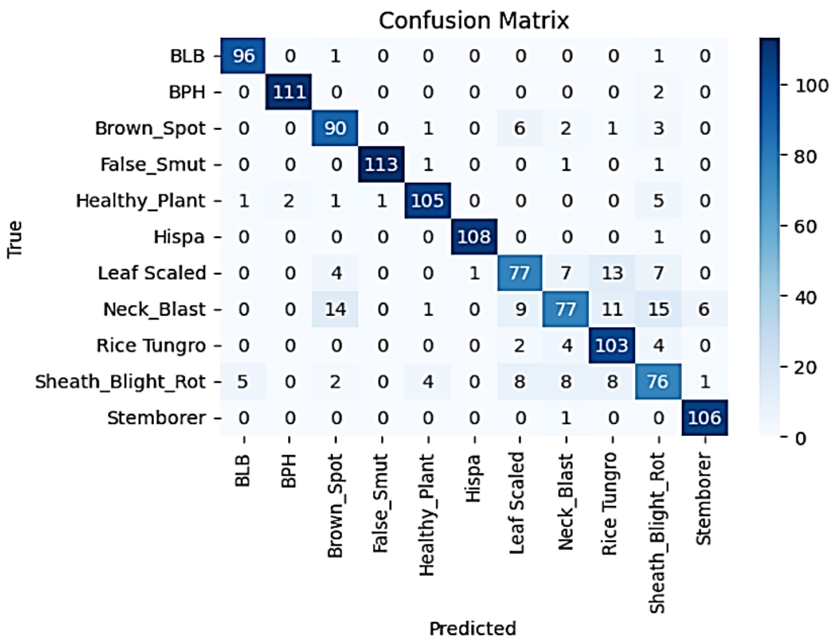


Figure 4.46: Confusion matrix of LiteViT_APSWWO

Figure 4.47 illustrates the ROC curve for LiteViT_APSWWO, offering insights into the trade-off between recall (True Positive Rate) and the False Positive Rate at various classification thresholds. Each point on this curve corresponds to a specific threshold value. Remarkably, the ROC-AUC scores for BLB, BPH, brown spot, false smut, healthy plant, hispa, leaf scaled, neck blast, tungro, sheath blight rot, and stemborer are exceptional, with values of 1.00, 1.00, 0.99, 1.00, 0.99, 1.00, 0.97, 0.95, 0.99, 0.95, and 1.00, respectively. These outstanding scores underscore the model's exceptional ability to accurately detect these specific diseases. BLB, BPH, false smut, hispa, and stemborer achieve the highest ROC-AUC scores, as depicted in Figure 4.50(d), while other classes also exhibit strong performance.

Figure 4.48 provides valuable insights into the balance between precision (the accuracy of positive predictions) and recall (the ability to identify all positive cases) as classification thresholds vary for LiteViT_APSWWO. Impressively, the Average Precision (AP) scores for BLB, BPH, brown spot, false smut, healthy plant, hispa, leaf scaled, neck blast, tungro, sheath blight rot, and stemborer are exceptional, with values of 0.99, 1.00, 0.92, 1.00, 0.97, 1.00, 0.78, 0.78, 0.91, 0.75, and 1.00, respectively. These remarkable AP scores highlight LiteViT_APSWWO's ability to effectively classify positive instances while minimizing false positives for BPH, false smut, hispa, and stemborer disease classes. However, the sheath blight rot class exhibits relatively lower AP scores compared to other classes, as indicated in Figure 4.50 (e).

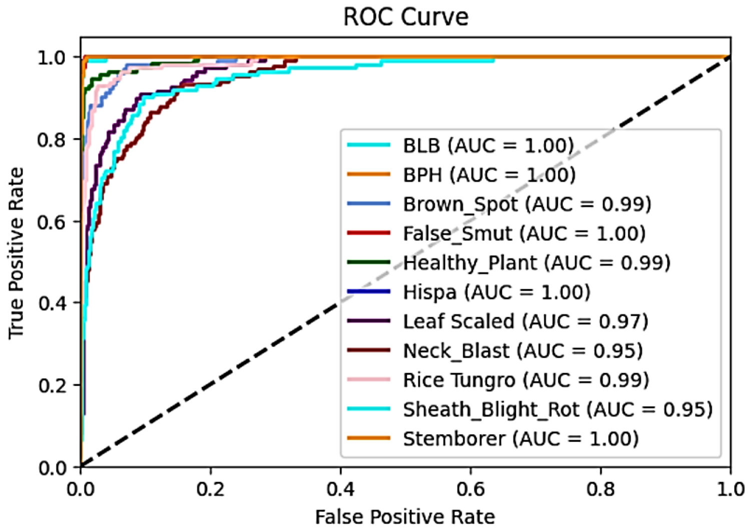


Figure 4.47: ROC curve of LiteViT_APSWWO

Figure 4.49 displays the learning curve for LiteViT_APSWWO. Early stopping was applied, restricting the model to only 20 epochs. During training, the training loss consistently decreased with each epoch, ultimately reaching a final value of 0.1097, indicating effective learning from the training data. However, the testing loss exhibited fluctuations throughout the epochs, concluding with a final testing loss of 0.5026. This suggests that while the model performed well on the training data, it faced challenges when generalizing to unseen data, as evidenced by the higher testing loss.

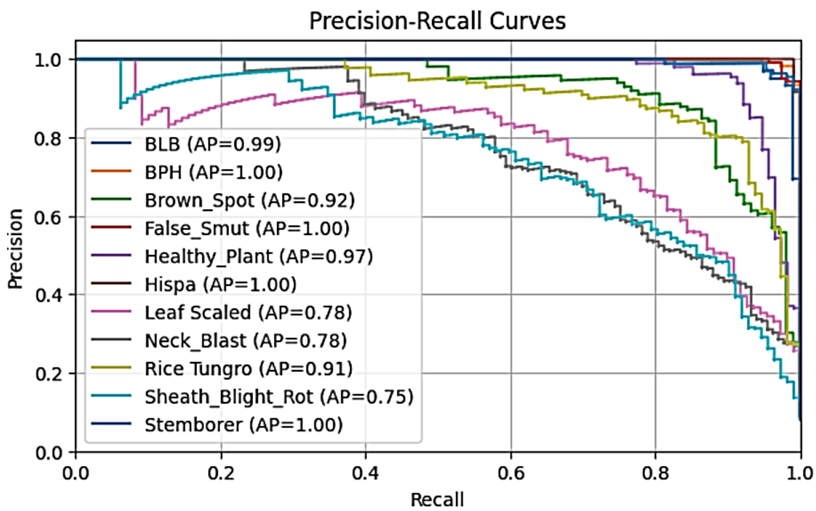


Figure 4.48: Precision-Recall curves of LiteViT_APSWWO

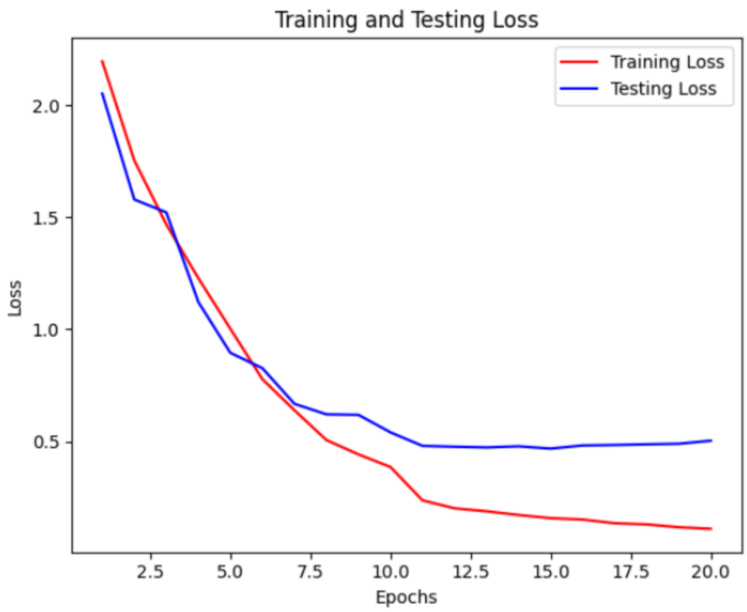
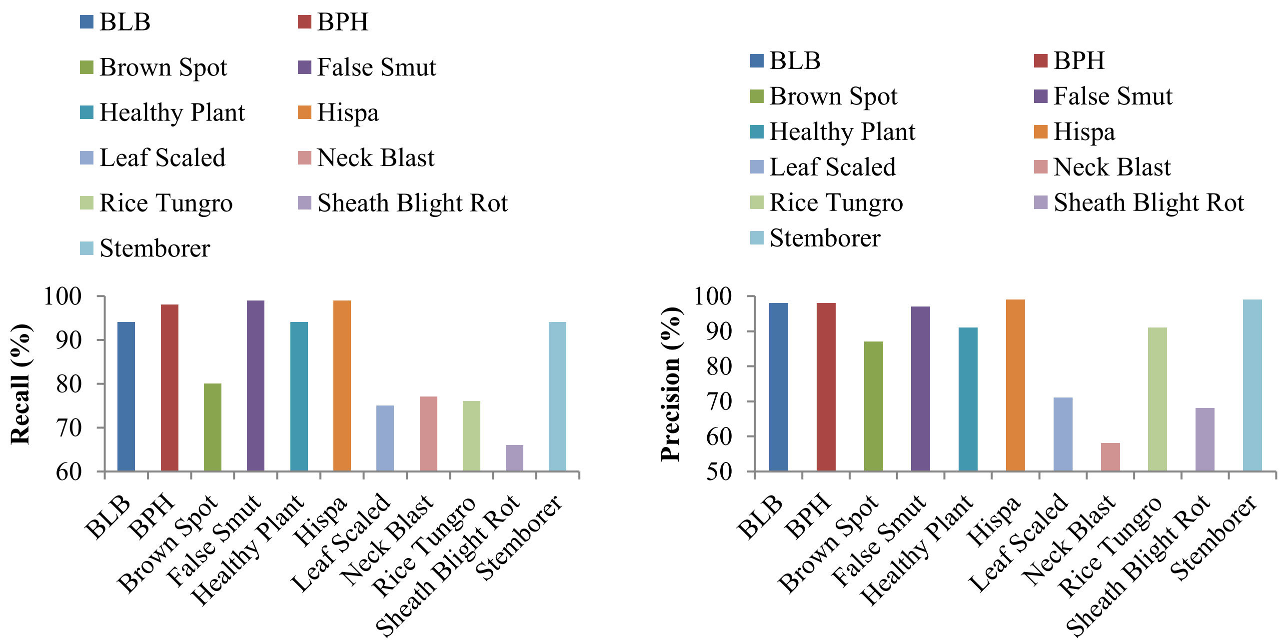


Figure 4.49: Learning curve of LiteViT_APSWWO

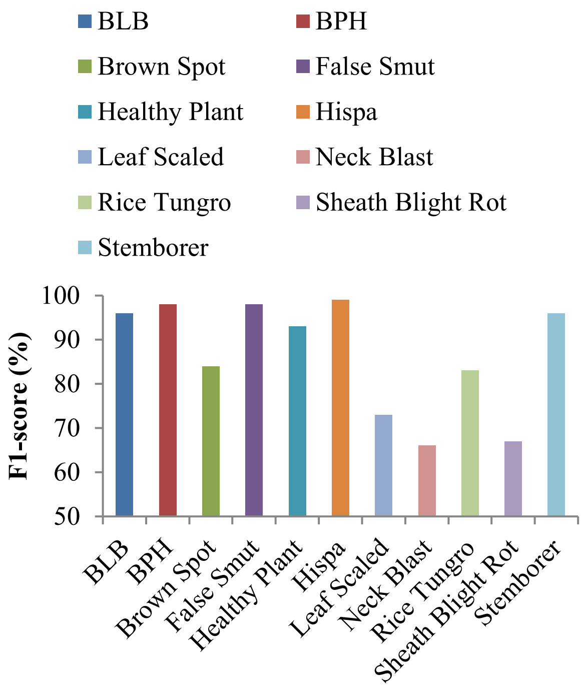
Table 4.10 provides an extensive overview of LiteViT_APSWWO's classification results, offering various performance metrics for each disease class. In the BLB class, the model achieves a recall, precision, F1-score, ROC_AUC score, and AP value of 94%, 98%, 96%, 1.00, and 0.99, respectively. Similarly, for the BPH class, these metrics are recorded at 98%, 98%, 98%, 1.00, and 1.00, respectively. The brown spot class exhibits lower metrics, with a recall of 80%, precision of 87%, F1-score of 84%, ROC_AUC score of 0.99, and AP value of 0.92. In contrast, the false smut class excels with a recall of 99%, precision of 97%, F1-score of 98%, ROC_AUC score of 1.00, and AP value of 1.00. The healthy plant class maintains strong performance, achieving values of 94% recall, 91% precision, 93% F1-score, ROC_AUC score of 0.99, and AP value of 0.97. Similarly, the Hispa class stands out with 99% recall, 99% precision, 99% F1-score, ROC_AUC score of 1.00, and AP value of 1.00. Leaf scaled exhibits a recall of 75%, precision of 71%, F1-score of 73%, ROC_AUC score of 0.97, and AP value of 0.78, while the neck blast class records metrics of 77% recall, 58% precision, 66% F1-score, ROC_AUC score of 0.95, and AP value of 0.78. The tungro class showcases a recall of 76%, precision of 91%, F1-score of 83%, ROC_AUC score of 0.99, and AP value of 0.91, while sheath blight rot class attains a recall of 66%, precision of 68%, F1-score of 67%, ROC_AUC score of 0.95, and AP value of 0.75. Finally, the stemborer class demonstrates strong performance with a recall of 94%, precision of 99%, F1-score of 96%, ROC_AUC score of 1.00, and AP value of 1.00. Figure 4.50 visually illustrates these performance metrics for each class using LiteViT_APSWWO, with false smut and Hispa achieving the highest recall at 98%, while sheath blight rot displays the lowest recall at 66%. Hispa leads in precision, with a score of 99%, while neck blast trails with the lowest precision at 58%. Hispa shines with a top F1-score of 99%, while neck blast lags behind with a 66% F1-score for the LiteViT_APSWWO model.

Table 4.10: Performance table of LiteViT_APSWWO

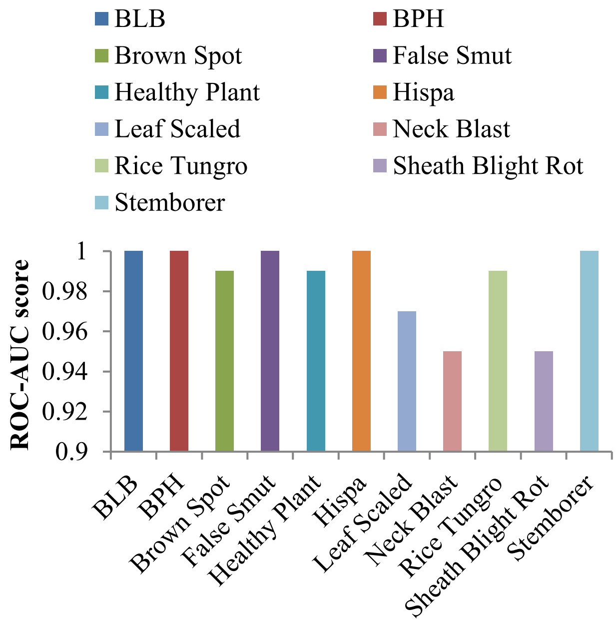
Class	Correctly Classified Instance	Incorrectly Classified Instance	Recall (%)	Precision (%)	F1-score (%)	ROC-AUC score	AP value
BLB	96	2	94	98	96	1.00	0.99
BPH	111	2	98	98	98	1.00	1.00
Brown Spot	90	13	80	87	84	0.99	0.92
False Smut	113	3	99	97	98	1.00	1.00
Healthy Plant	105	10	94	91	93	0.99	0.97
Hispa	108	1	99	99	99	1.00	1.00
Leaf Scaled	77	32	75	71	73	0.97	0.78
Neck Blast	77	56	77	58	66	0.95	0.78
Paddy Tungro	103	10	76	91	83	0.99	0.91
Sheath Blight Rot	76	36	66	68	67	0.95	0.75
Stemborer	106	1	94	99	96	1.00	1.00
Average	96.55	15.09	86.55	87	86.64	0.985	0.918



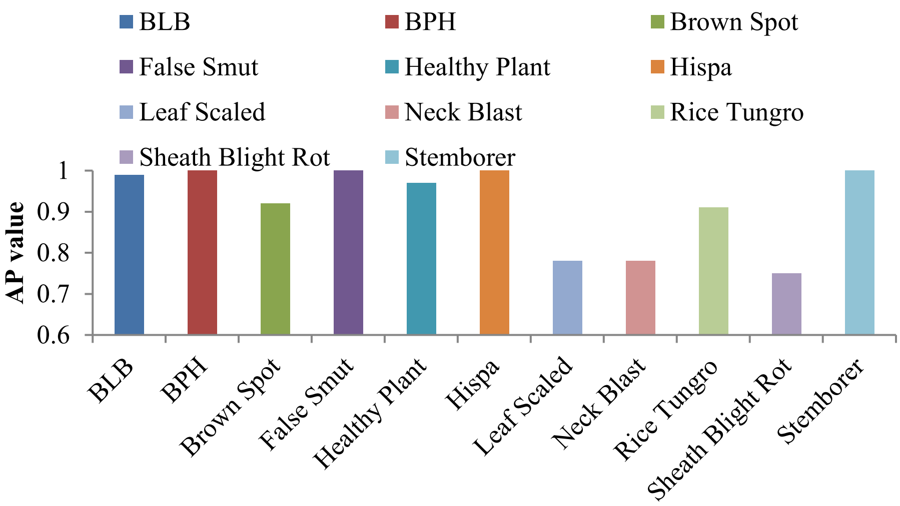
(a) Recall (b) Precision



(c) F1-score



(d) ROC-AUC score



(e) Average Precision value

Figure 4.50: Performance metrics for each class using LiteViT_APSWWO

4.3 Explainable AI Technique in Paddy Disease Detection

The result of Explainable AI technique using Grad-CAM is explored to gain insights into and interpret the inner workings of ANN models in paddy disease detection.

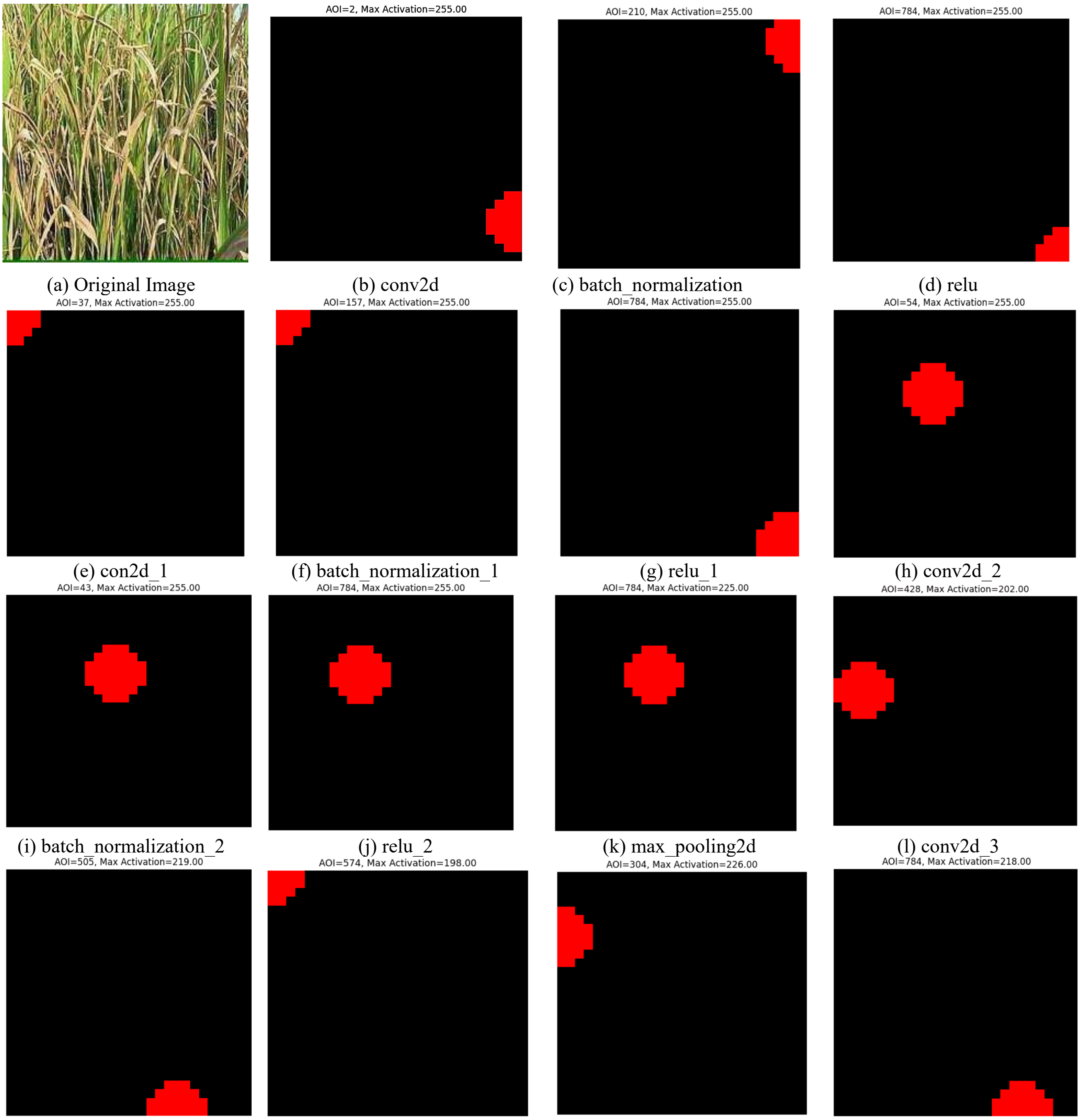
4.3.1 Convolutional Neural Network

In this section, the model interpretability technique Grad-CAM of CNN_Original, CNN_PSO, CNN_APSO, CNN_WWO, and CNN_APSWWO have been presented.

a) CNN_Original

This study has employed Grad-CAM techniques for each layer of the CNN model, applying them to each image in the dataset. Grad-CAM highlights the regions in an image responsible for a model's predictions. Figure 4.51 represents the Grad-CAM of each layer of CNN_Original. This study initiated the investigation by selecting a representative BLB image from the dataset. This image served as the input to the CNN_Original model, and aimed to dissect the model's attention and feature activation patterns at various layers. The red circle overlaid on each Grad-CAM visualization delineates the AOI (Area of Interest), representing the regions within the input image that have the most significant influence on the model's prediction. These regions are where the model focuses its attention when making decisions, while unimportant areas are colored black. The AOI, as indicated by the red circle, signifies the spatial region that the model identifies as being crucial for its prediction. It is calculated based on the intensity of feature activations within the convolutional layers. The size and location of the AOI vary across layers and provide insights into the model's hierarchical feature learning. Each Grad-CAM visualization includes a color intensity scale, where the maximum activation value is represented by the brightest color. The maximum activation value quantifies the strength of the model's attention within the AOI. A higher maximum activation value indicates that the model assigns more importance to the corresponding region when making its prediction. The Grad-CAM visualizations are generated using a threshold applied to the heatmap. Pixels in the heatmap with values above the specified threshold are considered part of the AOI and are enclosed by the red circle. Adjusting the threshold can influence the size and granularity of the AOI and provides a way to control the model's attention. This study has used a 0.3 threshold value for each layer of CNN_Original.

Table 4.11 presents layer-wise AOI and maximum activation values for CNN_Original. Initial layers detect fundamental patterns and edges with small AOIs (e.g., AOI = 45) and high maximum activations (Max Activation = 255). Subsequent layers refine features with larger AOIs (e.g., AOI = 600) and consistent maximum activations. Relu activations expand AOIs across features, maintaining high maximum activations. Convolutional layers focus on specific features, while batch_normalization widens AOIs. Attention mechanisms narrow AOIs, possibly highlighting specific details. Overall, layers maintain broad AOIs and high maximum activations, indicating their importance in feature extraction and classification. The variations in AOI and Max Activation values for each layer of the CNN_Original model occur due to the distinct roles and functions of each layer in the neural network. Each layer in the network is designed to capture specific features or patterns within the input image. As information progresses through the layers, the network's focus and feature extraction evolve. Early layers, such as "conv2d," "batch_normalization," and "relu" layers, are typically associated with low-level feature extraction. These layers detect fundamental patterns like edges, textures, and simple shapes in the input image. Deeper layers, including "conv2d_1," "batch_normalization_1," "relu_1," and beyond, engage in deeper-level feature learning. They capture increasingly complex and abstract features such as object parts, textures, and object compositions. Neural networks are inherently hierarchical, with low-level layers detecting basic features and deeper layers capturing more intricate ones. This hierarchical feature learning results in varying AOI and Max Activation values as the network adapts to different levels of feature abstraction. Batch normalization and pooling layers affect the distribution of feature activations. Batch normalization helps maintain stable activations, while pooling layers downsample and aggregate information. These layers influence the AOI and Max Activation values by enhancing or suppressing certain features, depending on the layer's position in the network.

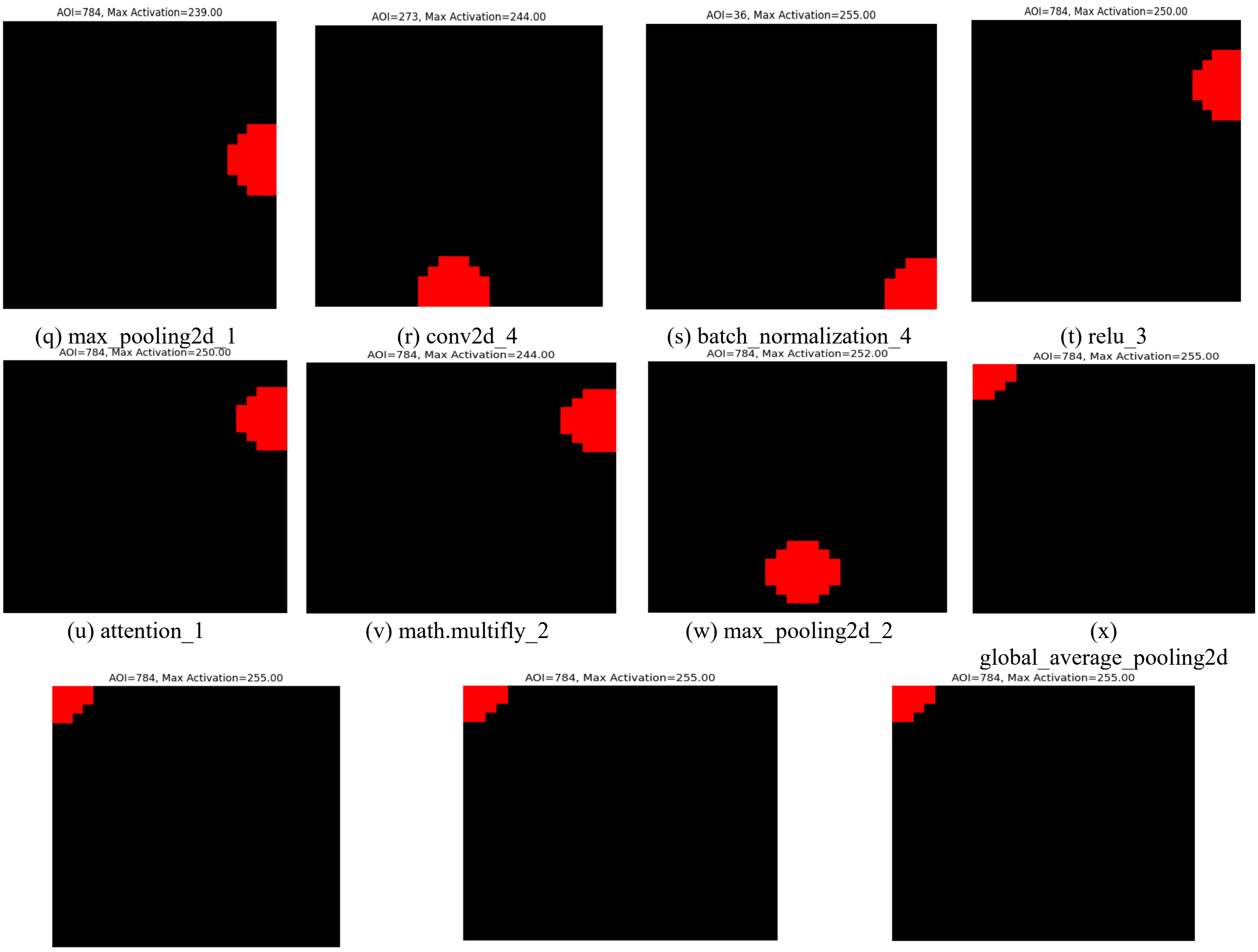


(a) Original Image (b) conv2d (c) batch_normalization (d) relu

(e) con2d_1 (f) batch_normalization_1 (g) relu_1 (h) conv2d_2

(i) batch_normalization_2 (j) relu_2 (k) max_pooling2d (l) conv2d_3

(m) batch_normalization_3 (n) math.multifly (o) attention (p) math.multifly_1



(q) max_pooling2d_1 (r) conv2d_4 (s) batch_normalization_4 (t) relu_3

(u) attention_1 (v) math.multifly_2 (w) max_pooling2d_2 (x) global_average_pooling2d

(y) dense (z) dropout (z1) dense_1

Figure 4.51: Grad-CAM of each layer of CNN_Original

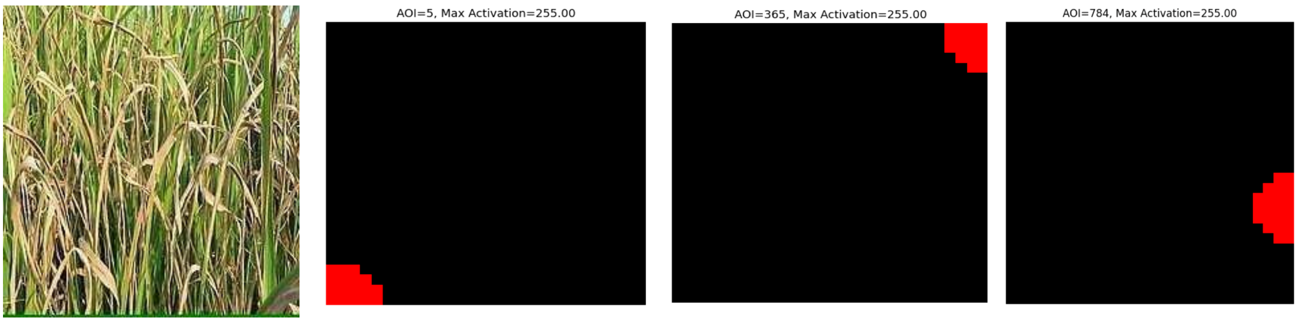
Table 4.11: Layer-wise Grad-CAM Values for CNN_Original

Layer	**AOI**	**Max activation**
conv2d	2	255
batch_normalization	210	255
relu	784	255
con2d_1	37	255
batch_normalization_1	157	255
relu_1	784	255
conv2d_2	54	255
batch_normalization_2	43	255
relu_2	784	255
max_pooling2d	784	225
conv2d_3	428	202

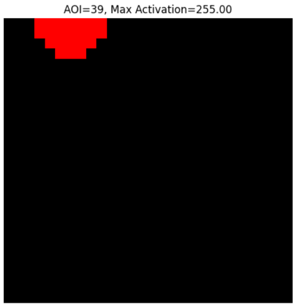
batch_normalization_3	505	219
math.multifly	574	198
attention	304	226
math.multifly_1	784	218
max_pooling2d_1	784	239
conv2d_4	273	244
batch_normalization_4	36	255
relu_3	784	250
attention_1	784	250
math.multifly_2	784	244
max_pooling2d_2	784	252
global_average_pooling2d	784	255
dense	784	255
dropout	784	255
dense_1	784	255

b) CNN_PSO

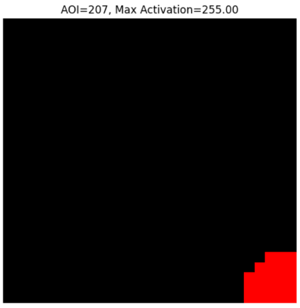
This study utilizes Grad-CAM techniques to analyze each layer of the CNN_PSO model, providing insights into its decision-making process for image classification. Figure 4.52 presents Grad-CAM visualizations for each CNN_PSO layer, starting with a representative BLB image. Red circles highlight Areas of Interest (AOIs), indicating crucial prediction regions, with size and location varying across layers. Grad-CAM images incorporate a color intensity scale representing maximum activation values, with a consistent threshold of 0.3 applied. Table 4.12 displays layer-wise AOI and maximum activation values for CNN_PSO. Initial layers focus on specific low-level features, while subsequent layers refine features over broader regions, maintaining strong activations. Convolutional layers identify specific patterns, with batch_normalization enhancing feature representation. Max-pooling layers down-sample features while preserving important information. Later layers contribute to mid-level and high-level feature extraction, with attention mechanisms modulating feature importance. Overall, layers exhibit varying AOIs and activation levels, highlighting their roles in feature extraction and classification.



(a) Original Image (b) conv2d (c) batch_normalization (d) relu

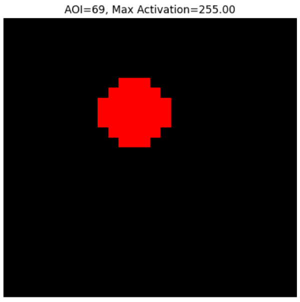


(e) con2d_1

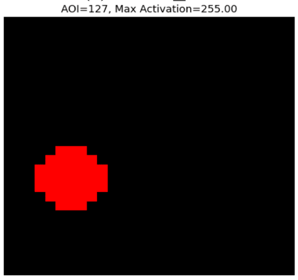


(f) batch_normalization_1

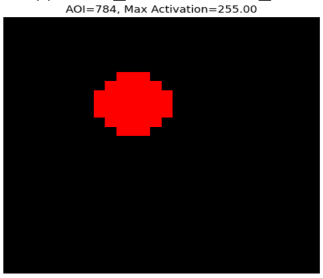
(g) relu_1



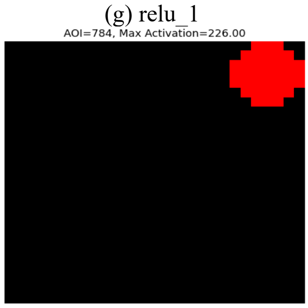
(h) conv2d_2



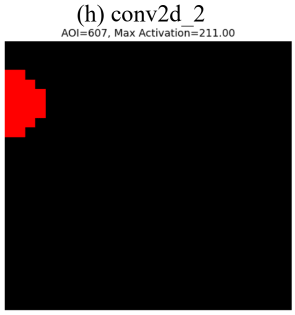
(i) batch_normalization_2



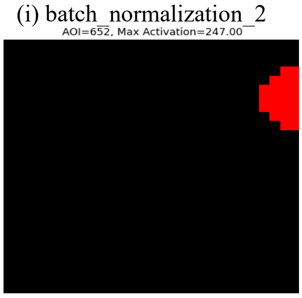
(j) relu_2



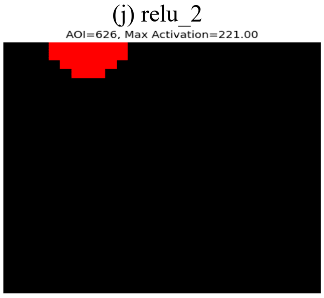
(k) max_pooling2d



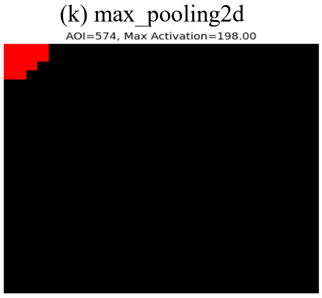
(l) conv2d_3



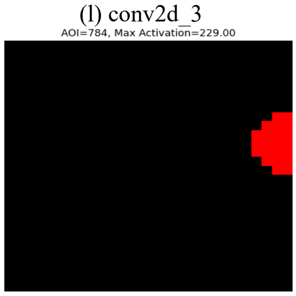
(m) batch_normalization_3



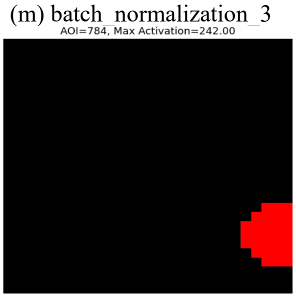
(n) math.multifly



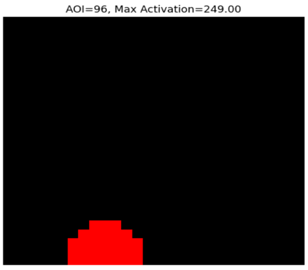
(o) attention



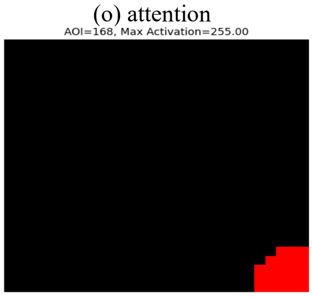
(p) math.multifly_1



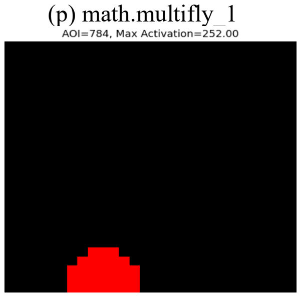
(q) max_pooling2d_1



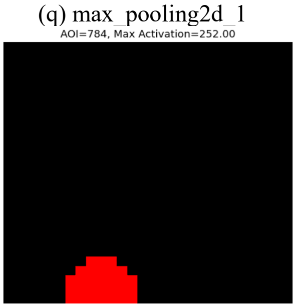
(r) conv2d_4



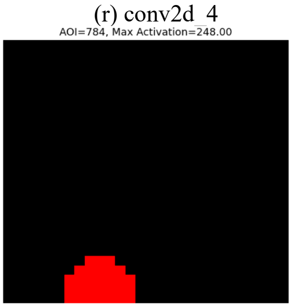
(s) batch_normalization_4



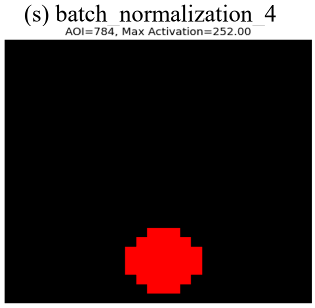
(t) relu_3



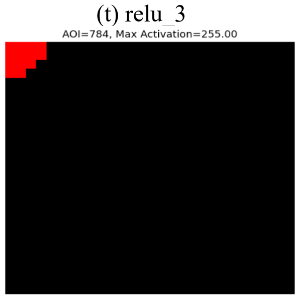
(u) attention_1



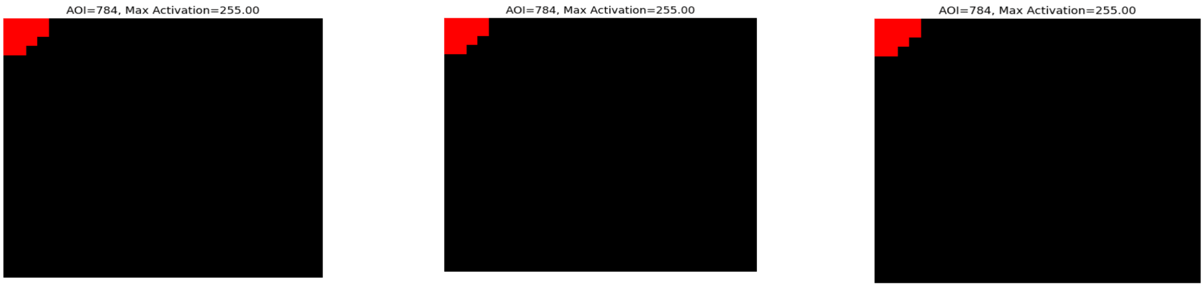
(v) math.multifly_2



(w) max_pooling2d_2



(x)
global_average_pooling2d



(y) dense (z) dropout (z1) dense_1

Figure 4.52: Grad-CAM of each layer of CNN_PSO

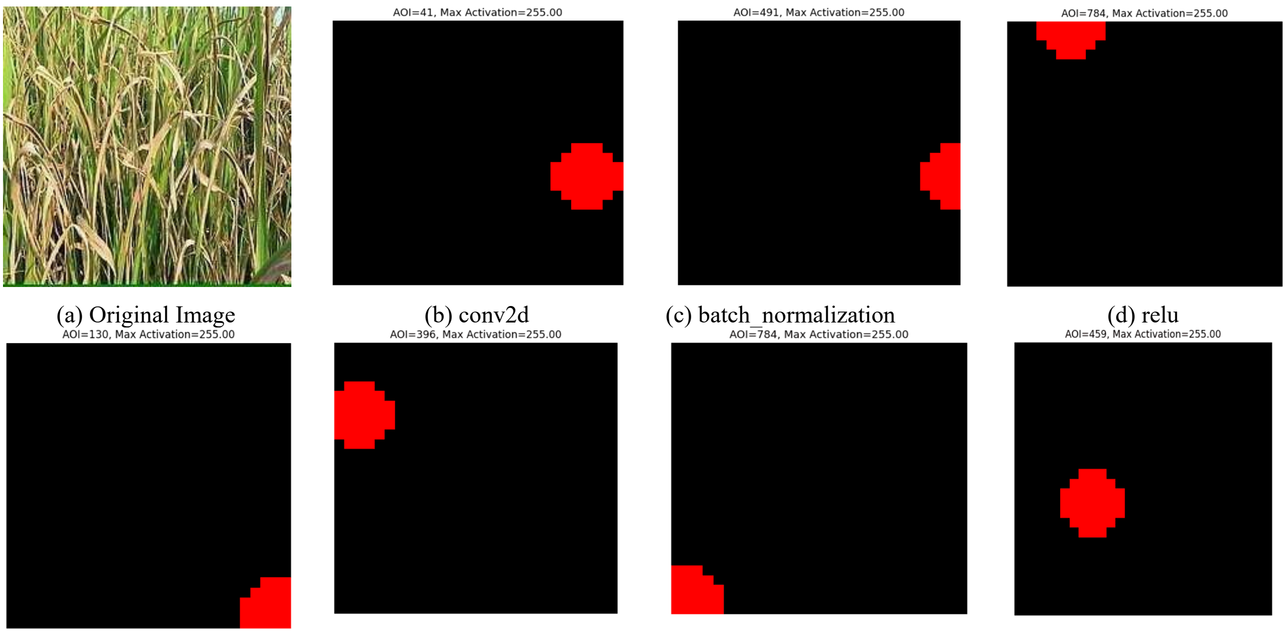
Table 4.12: Layer wise Grad-CAM value of CNN_PSO

Layer	**AOI**	**Max activation**
conv2d	5	255
batch_normalization	365	255
relu	784	255
con2d_1	39	255
batch_normalization_1	207	255
relu_1	784	255
conv2d_2	69	255
batch_normalization_2	127	255
relu_2	784	255
max_pooling2d	784	226
conv2d_3	607	211
batch_normalization_3	652	247
math.multifly	626	221
attention	574	198
math.multifly_1	784	229
max_pooling2d_1	784	242
conv2d_4	96	249
batch_normalization_4	168	255
relu_3	784	252
attention_1	784	252
math.multifly_2	784	248
max_pooling2d_2	784	255
global_average_pooling2d	784	255
dense	784	255
dropout	784	255
dense_1	784	255

The variation in AOI and maximum activation values across different layers of the CNN_PSO model reflects the distinct roles and feature extraction capabilities of each layer. Layers closer to the input, such as 'conv2d,' focus on local patterns within smaller AOIs, capturing fine-grained details with high maximum activation. As we progress deeper into the network, layers like 'attention_1' operate on larger AOIs, indicating their involvement in global feature learning and decision-making. The shifting position of the red circle in Grad-CAM visualizations signifies how the model dynamically allocates attention to different regions of the input image based on the layer's function. These variations collectively illustrate the hierarchical nature of feature extraction, where low-level layers emphasize local details, and deeper layers emphasize global patterns, ultimately contributing to the model's classification decisions.

c) CNN_APSO

This study utilizes Grad-CAM techniques to scrutinize each layer of the CNN_APSO model, unveiling its decision-making process in image classification. Figure 4.53 presents Grad-CAM visualizations for each CNN_APSO layer, starting with a representative BLB image. Red circles highlight Areas of Interest (AOIs), indicating crucial prediction regions, with size and location varying across different layers. Grad-CAM images incorporate a color intensity scale representing maximum activation values, with a consistent threshold of 0.3 applied. Table 4.13 provides layer-wise AOI and maximum activation values for CNN_APSO. Initial layers focus on specific low-level features, while subsequent layers refine features over broader regions, maintaining strong activations. The final dense_1 layer demonstrates an AOI covering all pixels, emphasizing a holistic view of the image essential for model predictions, with consistent high activation.



(a) Original Image (b) conv2d (c) batch_normalization (d) relu

(e) con2d_1 (f) batch_normalization_1 (g) relu_1 (h) conv2d_2

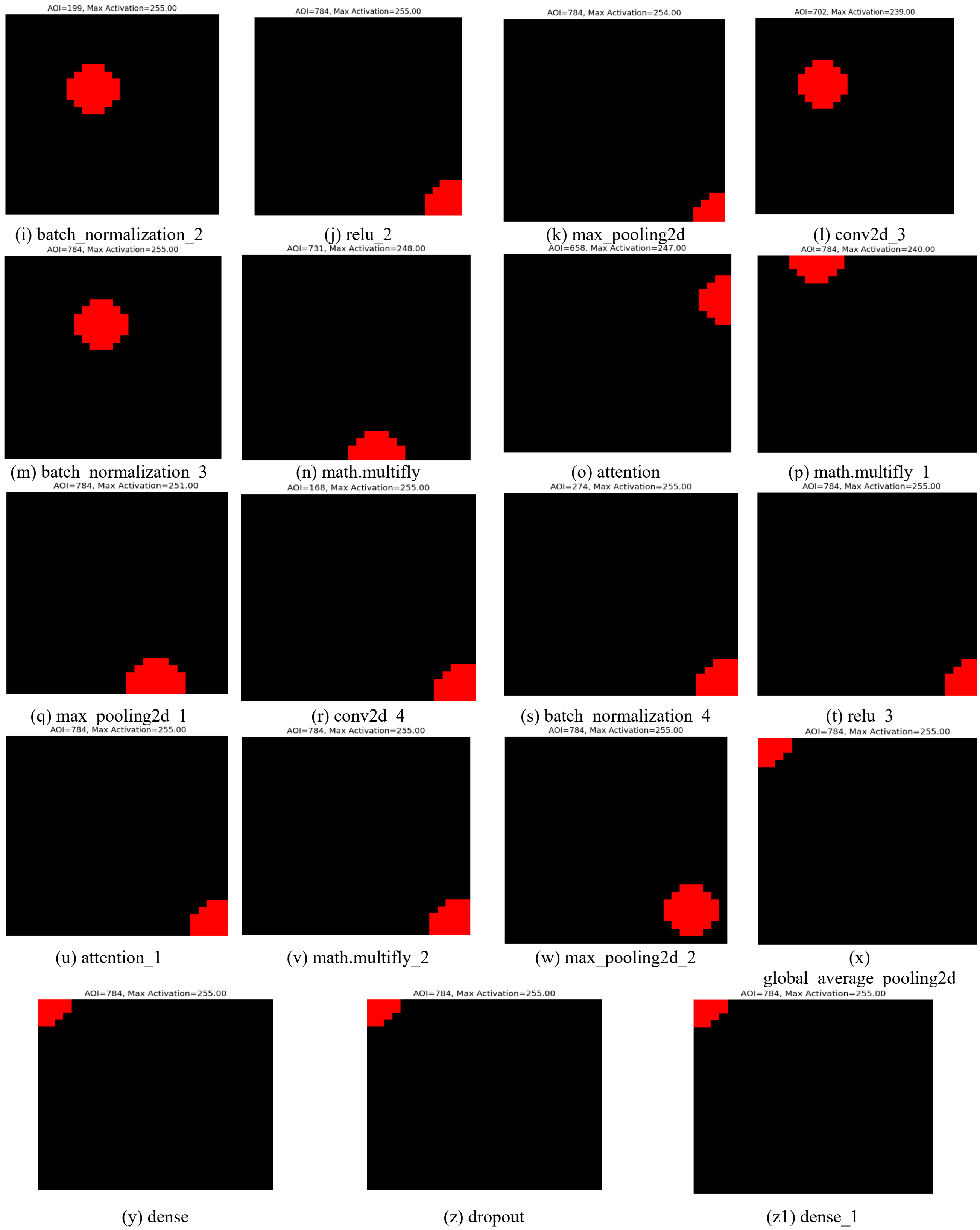


Figure 4.53: Grad-CAM of each layer of CNN_APSO

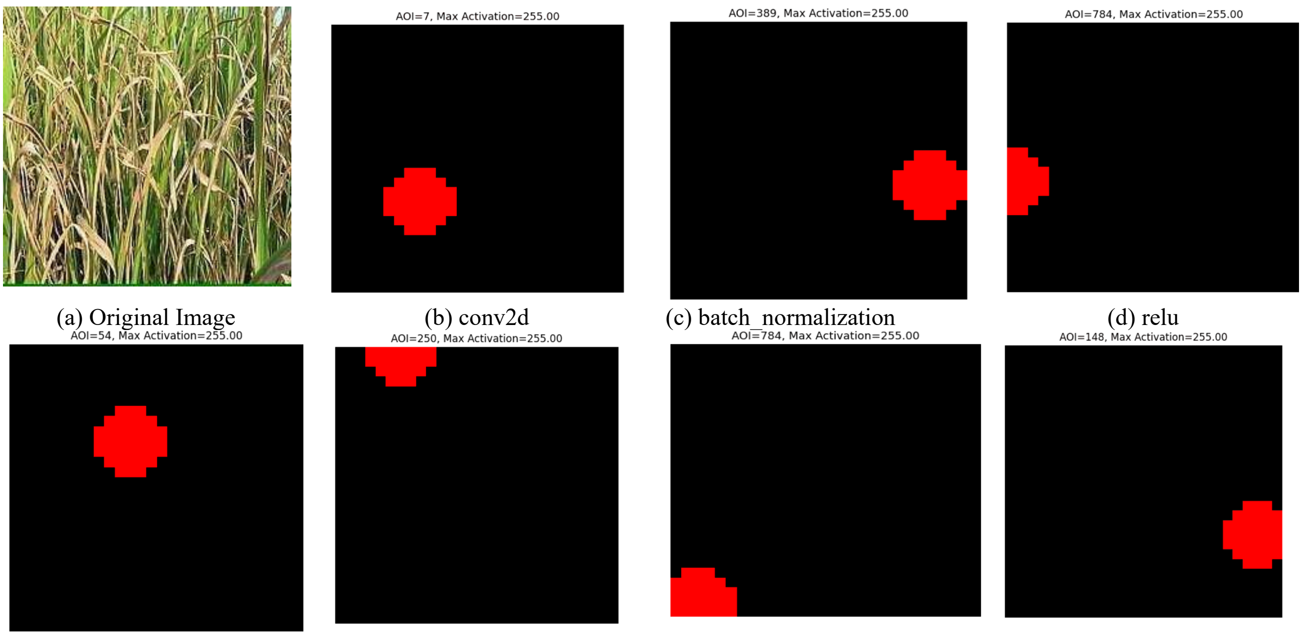
Table 4.13: Layer wise value of CNN_APSO

Layer	AOI	Max activation
conv2d	41	255
batch_normalization	491	255
relu	784	255
con2d_1	130	255
batch_normalization_1	396	255
relu_1	784	255
conv2d_2	459	255
batch_normalization_2	199	255
relu_2	784	255
max_pooling2d	784	254
conv2d_3	702	239
batch_normalization_3	784	255
math.multifly	731	248
attention	658	247
math.multifly_1	784	240
max_pooling2d_1	784	251
conv2d_4	168	255
batch_normalization_4	274	255
relu_3	784	255
attention_1	784	255
math.multifly_2	784	255
max_pooling2d_2	784	255
global_average_pooling2d	784	255
dense	784	255
dropout	784	255
dense_1	784	255

The variation in AOI (Area of Interest) and maximum activation values across different layers of the CNN_APSO model can be attributed to the model's hierarchical feature learning process. As the input image traverses through the network's convolutional and pooling layers, each layer extracts and emphasizes different features and patterns. Consequently, this leads to variations in the regions of interest and the intensity of attention assigned to them. The dynamic nature of CNN_APSO's architecture allows it to progressively focus on different aspects of the input image as it traverses deeper layers, resulting in varying AOIs and maximum activation values. The distinct positioning of the red circle in the Grad-CAM visualizations further reflects the layer-specific attention patterns, highlighting that different layers prioritize different regions within the image based on their relevance to the classification task. This diversity in AOI and activation patterns showcases the network's adaptability and capacity to learn intricate features at multiple levels of abstraction.

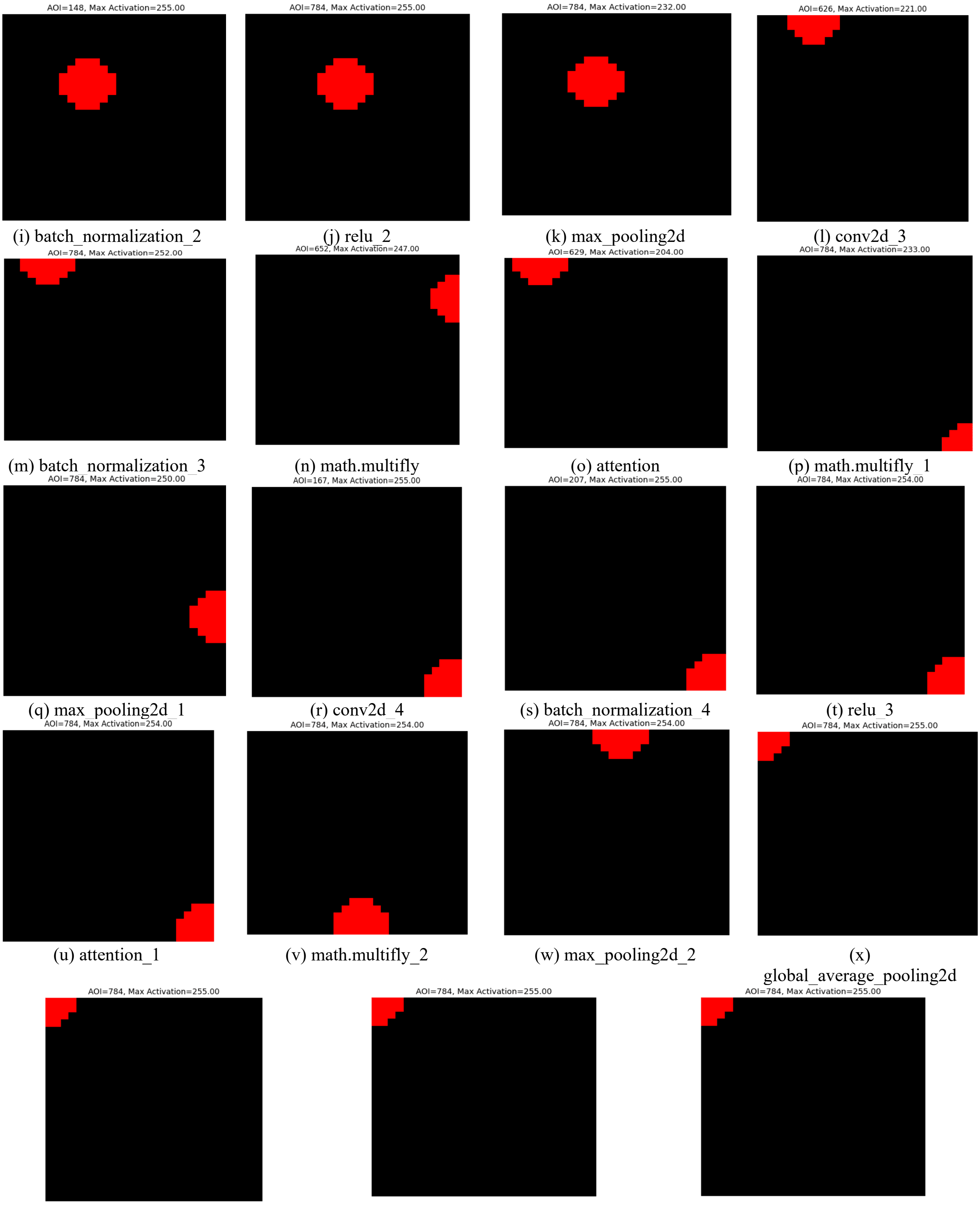
d) CNN_WWO

This study employs Grad-CAM techniques to dissect each layer of the CNN_WWO model, unveiling its decision-making process in image classification. Figure 4.54 presents Grad-CAM visualizations corresponding to individual CNN_WWO layers. The inquiry begins with the selection of a representative BLB image from the dataset, serving as input to the CNN_WWO model. Grad-CAM images feature distinctive red circles, emphasizing the Areas of Interest (AOIs), shedding light on regions pivotal to the model's decision-making. These AOIs, depicted in bold red, signify spatial regions of paramount importance in prediction. Characteristics of AOIs, such as size and location, vary across different layers, offering valuable insights into the model's hierarchical feature learning. Within each Grad-CAM image, a color intensity scale reflects the maximum activation value, with brighter colors denoting heightened model attention within the AOI. The Grad-CAM visualizations incorporate a consistent heatmap threshold of 0.3, facilitating the identification of AOI pixels, while less critical areas are visually represented in black. This thresholding approach ensures meticulous control over AOI size and granularity across various CNN_WWO layers.



(a) Original Image (b) conv2d (c) batch_normalization (d) relu

(e) con2d_1 (f) batch_normalization_1 (g) relu_1 (h) conv2d_2



(i) batch_normalization_2 (j) relu_2 (k) max_pooling2d (l) conv2d_3

(m) batch_normalization_3 (n) math.multifly (o) attention (p) math.multifly_1

(q) max_pooling2d_1 (r) conv2d_4 (s) batch_normalization_4 (t) relu_3

(u) attention_1 (v) math.multifly_2 (w) max_pooling2d_2 (x) global_average_pooling2d

(y) dense (z) dropout (z1) dense_1

Figure 4.54: Grad-CAM of each layer of CNN_WWO

Table 4.14 provides a layer-wise analysis of CNN_WWO, detailing the model's characteristics in terms of AOI (Area of Interest) and maximum activation. The initial layers, such as 'conv2d,' 'batch_normalization,' and 'relu,' demonstrate early-stage feature extraction capabilities with small AOIs and robust maximum activations. Subsequent layers maintain high attention to specific features while contributing significantly to the decision-making process. As we progress through the network, variations in AOI and maximum activation highlight the dynamic nature of feature extraction and hierarchical learning within the model. Finally, fully connected layers maintain consistent AOIs and maximum activations, emphasizing their stable contributions to the model's predictions.

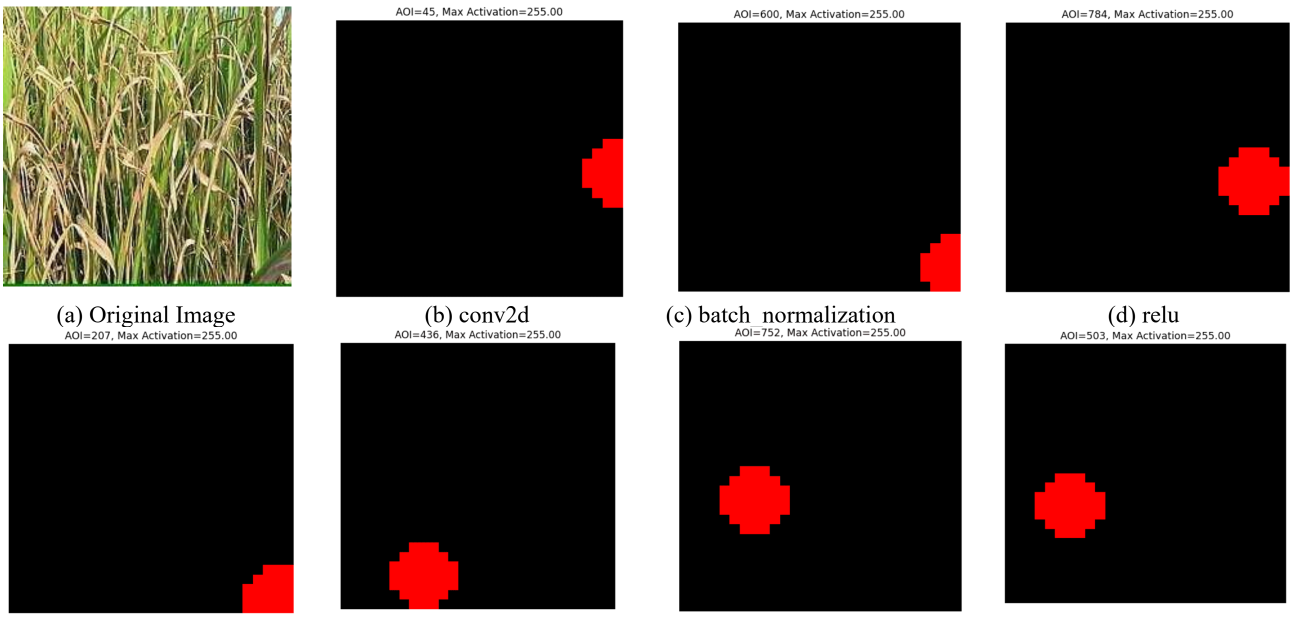
Table 4.14: Layer wise value of CNN_WWO

Layer	AOI	Max activation
conv2d	7	255
batch_normalization	389	255
relu	784	255
con2d_1	54	255
batch_normalization_1	250	255
relu_1	784	255
conv2d_2	148	255
batch_normalization_2	148	255
relu_2	784	255
max_pooling2d	784	232
conv2d_3	626	221
batch_normalization_3	784	252
math.multifly	652	247
attention	629	204
math.multifly_1	784	233
max_pooling2d_1	784	250
conv2d_4	167	255
batch_normalization_4	207	255
relu_3	784	254
attention_1	784	254
math.multifly_2	784	254
max_pooling2d_2	784	254
global_average_pooling2d	784	255
dense	784	255
dropout	784	255
dense_1	784	255

The varying values of AOI (Area of Interest) and maximum activation across the layers of the CNN_WWO model are indicative of the model's evolving focus and feature extraction hierarchy. In earlier layers, such as 'conv2d,' 'batch_normalization,' and 'relu,' the model exhibits a broad AOI and high maximum activation, suggesting a comprehensive grasp of fundamental features. As we progress deeper into the network, layers like 'conv2d_2' and 'conv2d_3' display more refined AOIs, indicating a shift towards more specialized feature extraction. The divergence in AOI and maximum activation highlights the layer-specific attention distribution within the model. The red circle's varying positions in the Grad-CAM visualizations signify the specific regions where the model concentrates its attention for each layer, adapting its focus as it processes the input image. This dynamic behavior underlines the model's capacity to adapt and learn hierarchical features, progressively honing its attention towards areas critical for accurate predictions.

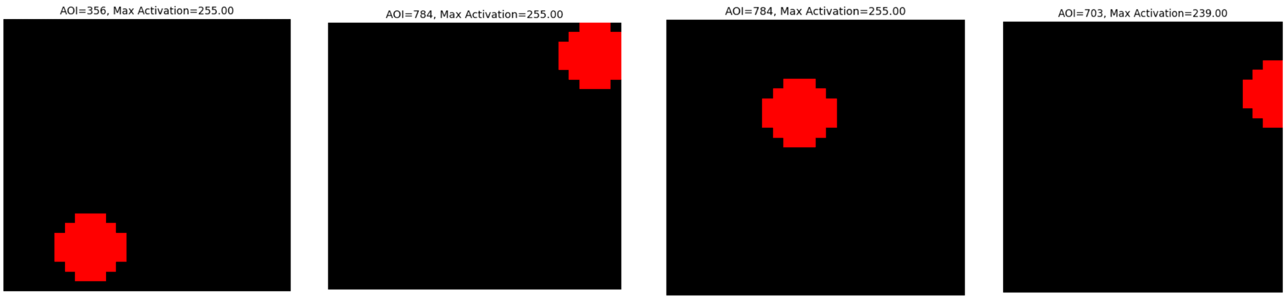
e) CNN_APSWWO

This study uses Grad-CAM to analyze CNN_APSWWO layers' decision-making in image classification. Figure 4.55 shows Grad-CAM for each layer, highlighting crucial Areas of Interest (AOIs) in red. AOIs vary in size and location, indicating hierarchical feature learning. A consistent 0.3 heatmap threshold ensures precise AOI identification. In Table 4.15, CNN_APSWWO layers exhibit varying Area of Interest (AOI) and Maximum Activation values, revealing nuanced information processing and prioritization. Initial layers like 'conv2d' focus on specific image regions with an AOI of 45 and a Max Activation of 255. 'batch_normalization' expands to broader features (AOI 600) while 'relu' maintains consistency across the input (AOI 784). Subsequent layers alternate between localized processing and wider engagement, reflecting complex feature extraction dynamics. 'conv2d_3' shows a more selective focus (AOI 703), while later layers maintain uniform AOI and Max Activation values (784 and 255), suggesting a higher level of abstraction and comprehensive understanding in decision-making.

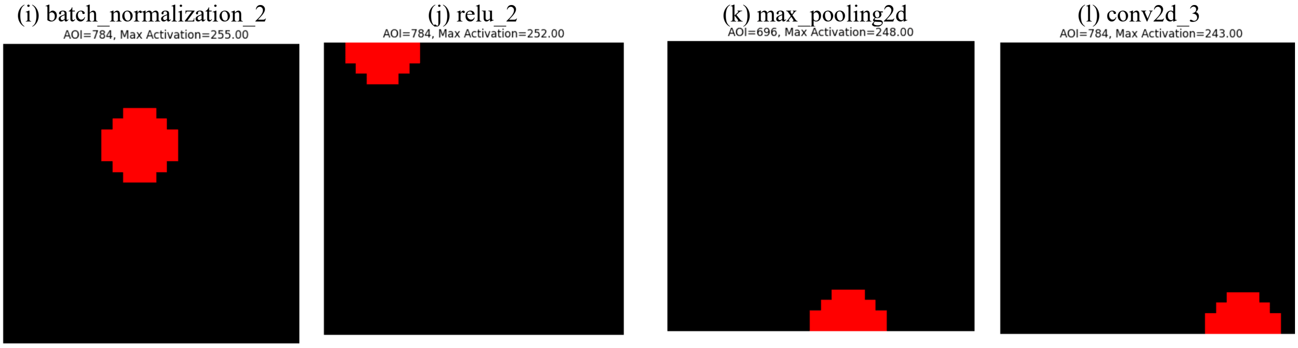


(a) Original Image (b) conv2d (c) batch_normalization (d) relu

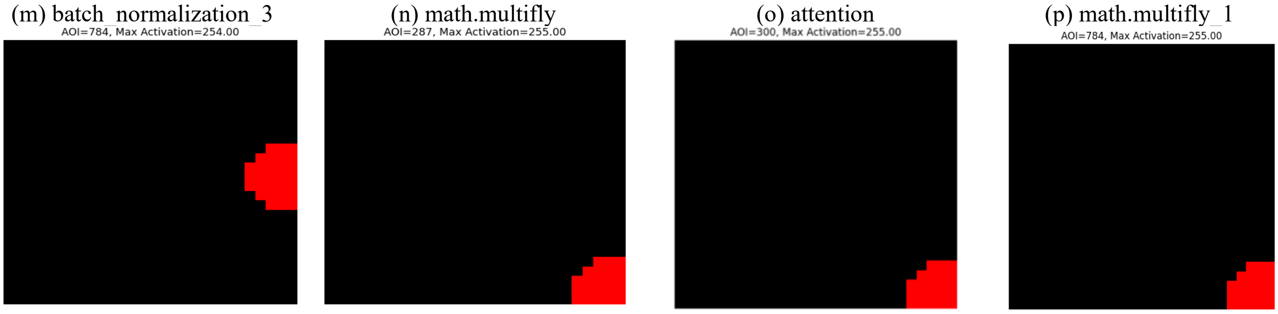
(e) con2d_1 (f) batch_normalization_1 (g) relu_1 (h) conv2d_2



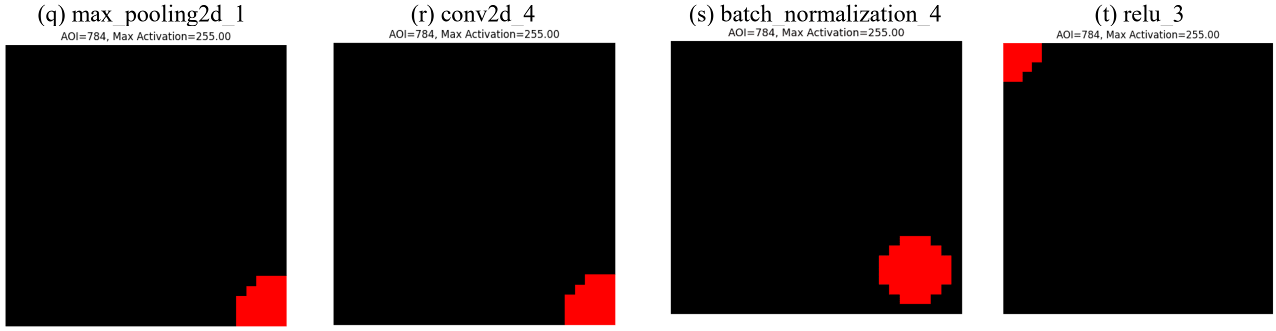
(i) batch_normalization_2 (j) relu_2 (k) max_pooling2d (l) conv2d_3



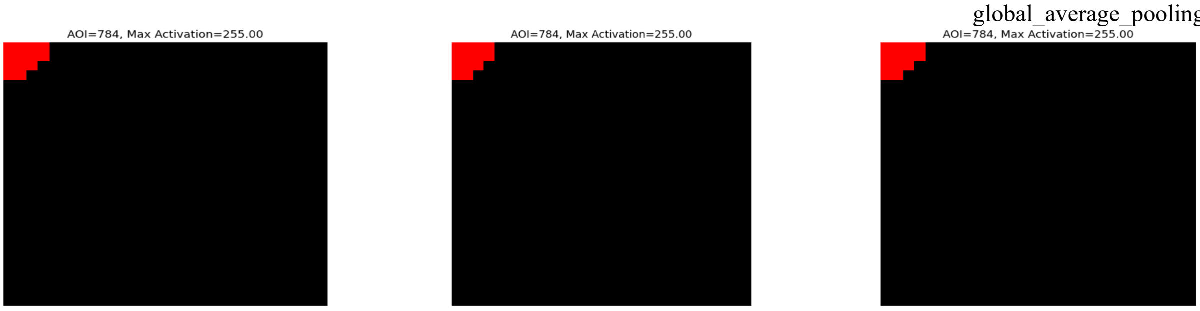
(m) batch_normalization_3 (n) math.multifly (o) attention (p) math.multifly_1



(q) max_pooling2d_1 (r) conv2d_4 (s) batch_normalization_4 (t) relu_3



(u) attention_1 (v) math.multifly_2 (w) max_pooling2d_2 (x) global_average_pooling2d



(y) dense (z) dropout (z1) dense_1

Figure 4.55: Grad-CAM of each layer of CNN_APSWWO

Table 4.15: Layer wise value of CNN_APSWWO

Layer	AOI	Max activation
conv2d	45	255
batch_normalization	600	255
relu	784	255
con2d_1	207	255
batch_normalization_1	436	255
relu_1	752	255
conv2d_2	503	255
batch_normalization_2	356	255
relu_2	784	255
max_pooling2d	784	255
conv2d_3	703	239
batch_normalization_3	784	255
math.multifly	784	252
attention	696	248
math.multifly_1	784	243
max_pooling2d_1	784	254
conv2d_4	287	255
batch_normalization_4	300	255
relu_3	784	255
attention_1	784	255
math.multifly_2	784	255
max_pooling2d_2	784	255
global_average_pooling2d	784	255
dense	784	255
dropout	784	255
dense_1	784	255

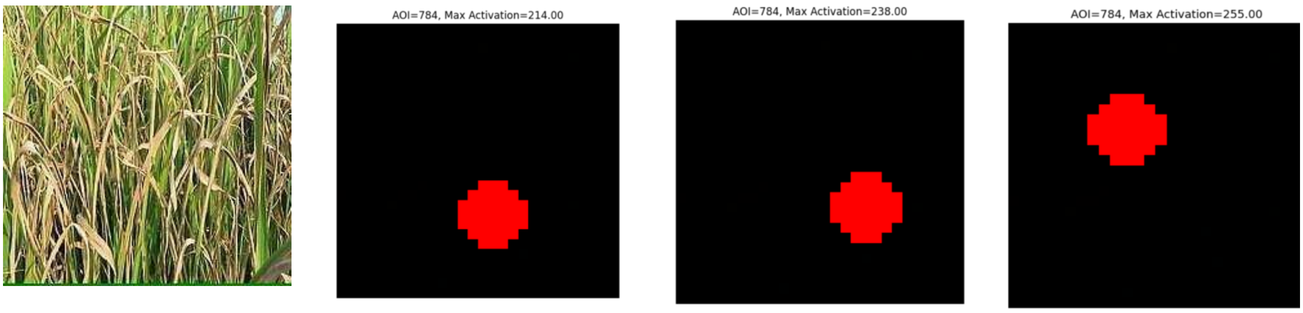
4.3.2 Light Weight Vision Transformer

In this section, the result obtained through the Grad-CAM of Light Weight Vision Transformer (LiteViT_Original, LiteViT_PSO, LiteViT_APSO, LiteViT_WWO, and LiteViT_APSWWO) models has been examined to provide visualizations of the model's internal operations and enhance understanding of its behavior.

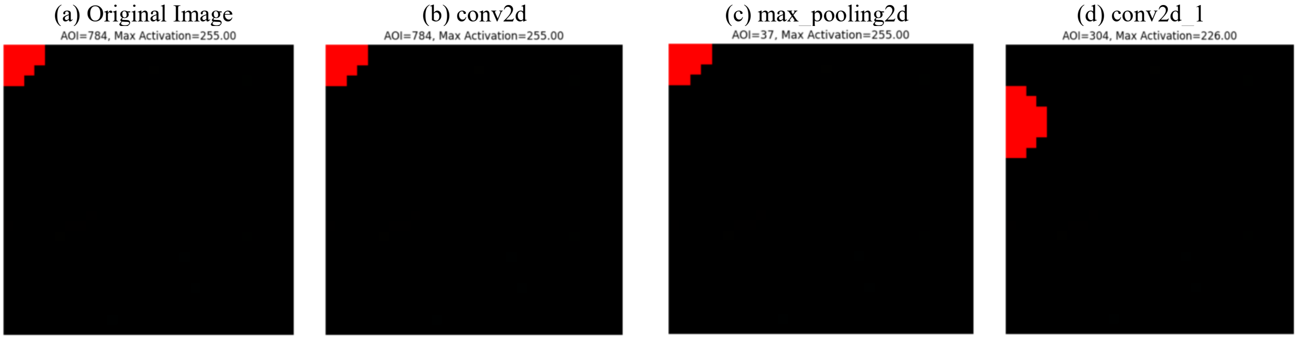
a) LiteViT_Original

In this study, Grad-CAM techniques were applied to each layer of the LiteViT_Original model, analyzing individual images from the dataset. Grad-CAM highlights critical regions in images that influence the model's predictions. By using Grad-CAM across different layers of

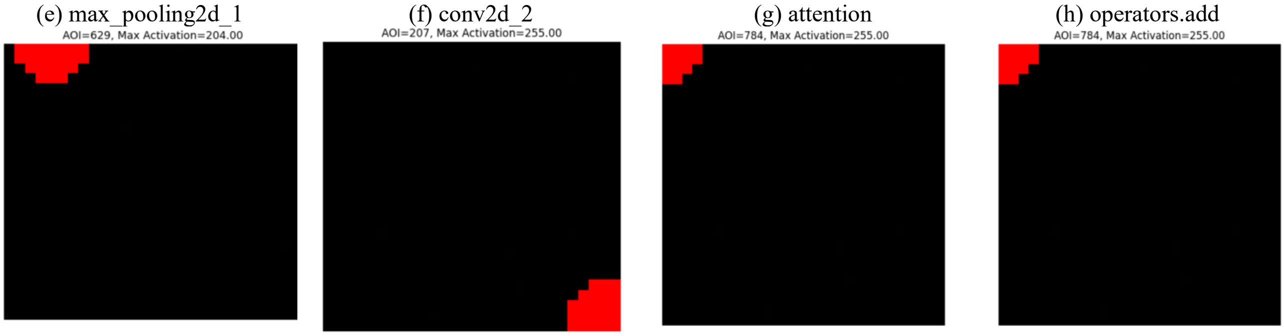
LiteViT_Original for each image, researchers gained insights into the model's decision-making process and hierarchical feature learning. Figure 4.56 displays Grad-CAM results for each layer, starting with a representative BLB image as input. Each visualization includes a red circle indicating the Area of Interest (AOI), where the model focuses its attention. A color intensity scale reflects the maximum activation value, indicating the model's focus strength within the AOI. A consistent threshold of 0.3 controlled the size and granularity of the AOI across all layers. Table 4.16 presents insights into LiteViT_Original's inner workings, detailing AOI and maximum activations per layer. Initially, attention is focused across 784 regions in Conv2D and subsequent layers, with varying maximum activations. Attention becomes more selective in later layers, notably in Attention and Operators.add_1 layers. However, the model consistently maintains a broad focus on the entire image in GlobalAveragePooling2D, Dense, Dropout, and Dense_1 layers. The variations in the number of AOI and their associated maximum activation values from layer to layer indicate that the LiteViT_Original model progressively extracts and weighs information from the input image. Layers with the same results might still influence different aspects of the model's decision-making process, or they might share similarities in terms of importance. The change in the red circle's location in each layer signifies the model's evolving focus on distinct regions of the image as it undergoes feature extraction and hierarchical decision-making.



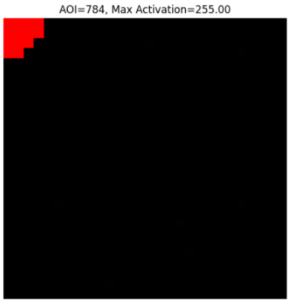
(a) Original Image (b) conv2d (c) max_pooling2d (d) conv2d_1



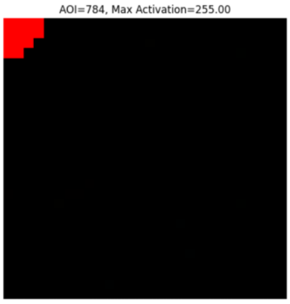
(e) max_pooling2d_1 (f) conv2d_2 (g) attention (h) operators.add



(i) attention_1 (j) operators.add_1 (k) global_average_pooling2d (l) dense



(m) dropout



(n) dense_1

Figure 4.56: Grad-CAM of each layer of LiteViT_Original

Table 4.16: Layer-wise Grad-CAM Values for LiteViT_Original

Layer	AOI	Max activation
conv2d	784	214
max_pooling2d	784	238
conv2d_1	784	255
max_pooling2d_1	784	255
conv2d_2	784	255
attention	37	255
operators.add	304	226
attention_1	629	204
operators.add_1	207	255
global_average_pooling2d	784	255
dense	784	255
dropout	784	255
dense_1	784	255

b) LiteViT_PSO

In this study, Grad-CAM is used to analyze LiteViT_PSO's decision-making across layers. Figure 4.57 visualizes these insights, highlighting Areas of Interest (AOIs) crucial for predictions. AOIs vary in size and location, indicating hierarchical feature learning. A consistent heatmap threshold of 0.3 controls AOI granularity. Table 4.17 details layer-wise AOI values and maximum activations, providing a comprehensive understanding of LiteViT_PSO's inner workings in image classification. In Table 4.17, the Grad-CAM analysis of LiteViT_PSO showcases the model's attention dynamics across layers. In "conv2d," a broad AOI of 784 pixels with a high maximum activation of 246 signifies important features. Subsequent layers like "max_pooling2d" maintain AOI but slightly reduce activation, indicating continued focus on significant regions. "Conv2d_1" sustains this focus with a consistent AOI and maximum activation of 255. Similarly, "attention" shifts to a smaller, focused AOI, while "operators.add" widens focus with a large AOI. "Global_average_pooling2d" and dense layers maintain a broad AOI, preparing for final classification. The red circles in Grad-CAM visualize AOI, reflecting evolving feature learning hierarchy.

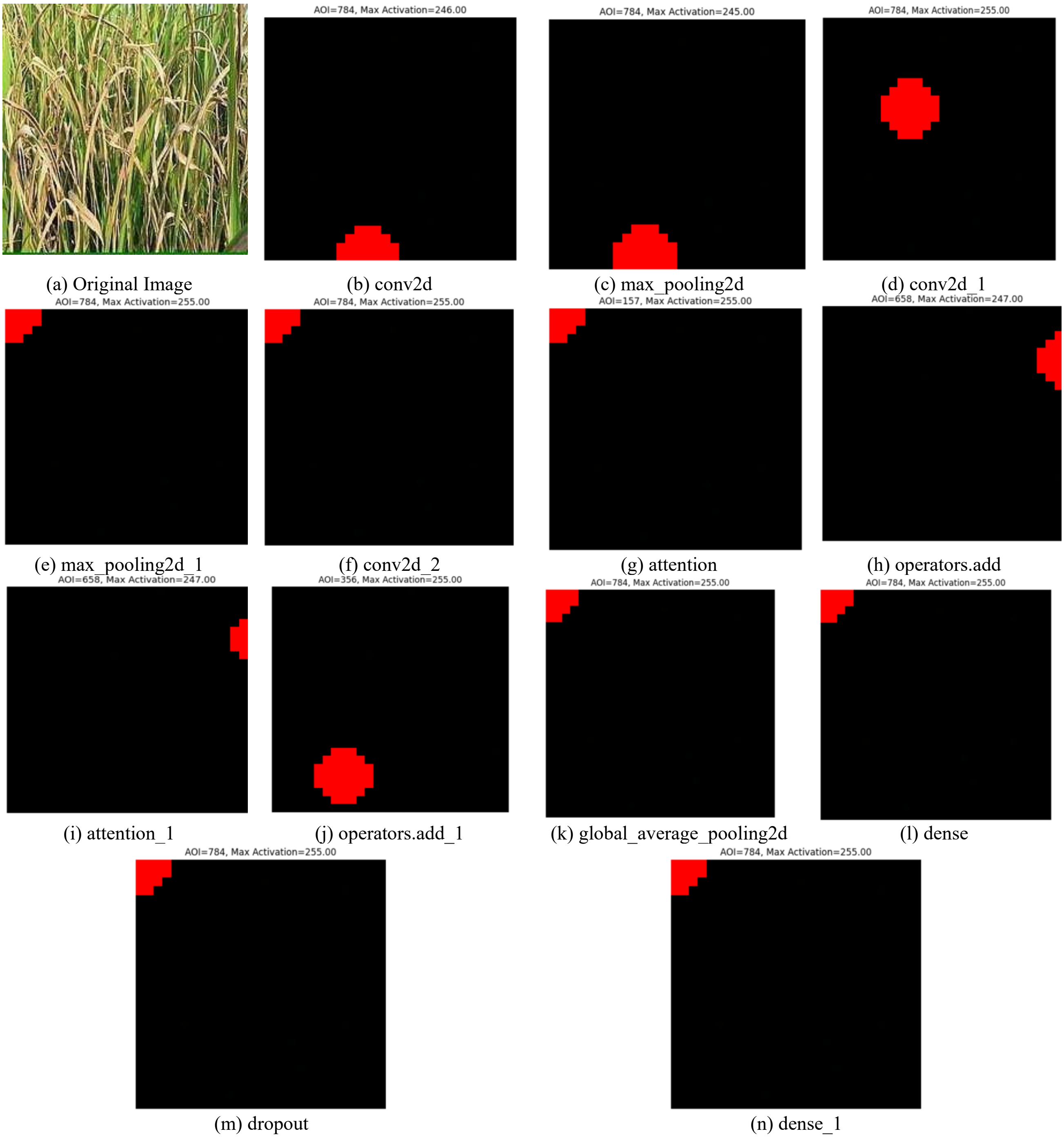


Figure 4.57: Grad-CAM of each layer of LiteViT_PSO

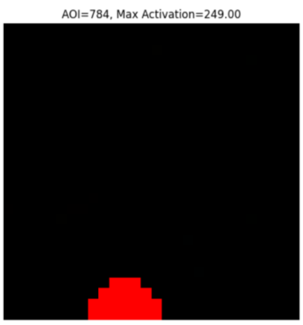
Table 4.17: Layer-wise Grad-CAM Values for LiteViT_PSO

Layer	AOI	Max activation
conv2d	784	246
max_pooling2d	784	245
conv2d_1	784	255
max_pooling2d_1	784	255
conv2d_2	784	255
attention	157	255
operators.add	658	255
attention_1	658	247
operators.add_1	356	255
global_average_pooling2d	784	255
dense	784	255
dropout	784	255
dense_1	784	255

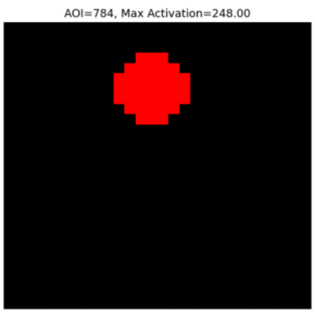
c) LiteViT_APSO

This study employs Grad-CAM techniques to analyze each layer of the LiteViT_APSO model, unraveling its decision-making process in image classification. Figure 4.58 presents Grad-CAM visualizations for each layer, starting with a representative BLB image as input. Red circles highlight Areas of Interest (AOIs), crucial for model decisions, with variations in size and location across layers. A color intensity scale reflects maximum activation, with brighter colors indicating higher attention. A consistent heatmap threshold of 0.3 controls AOI identification, ensuring granularity.

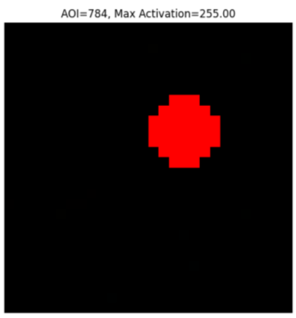

(a) Original Image



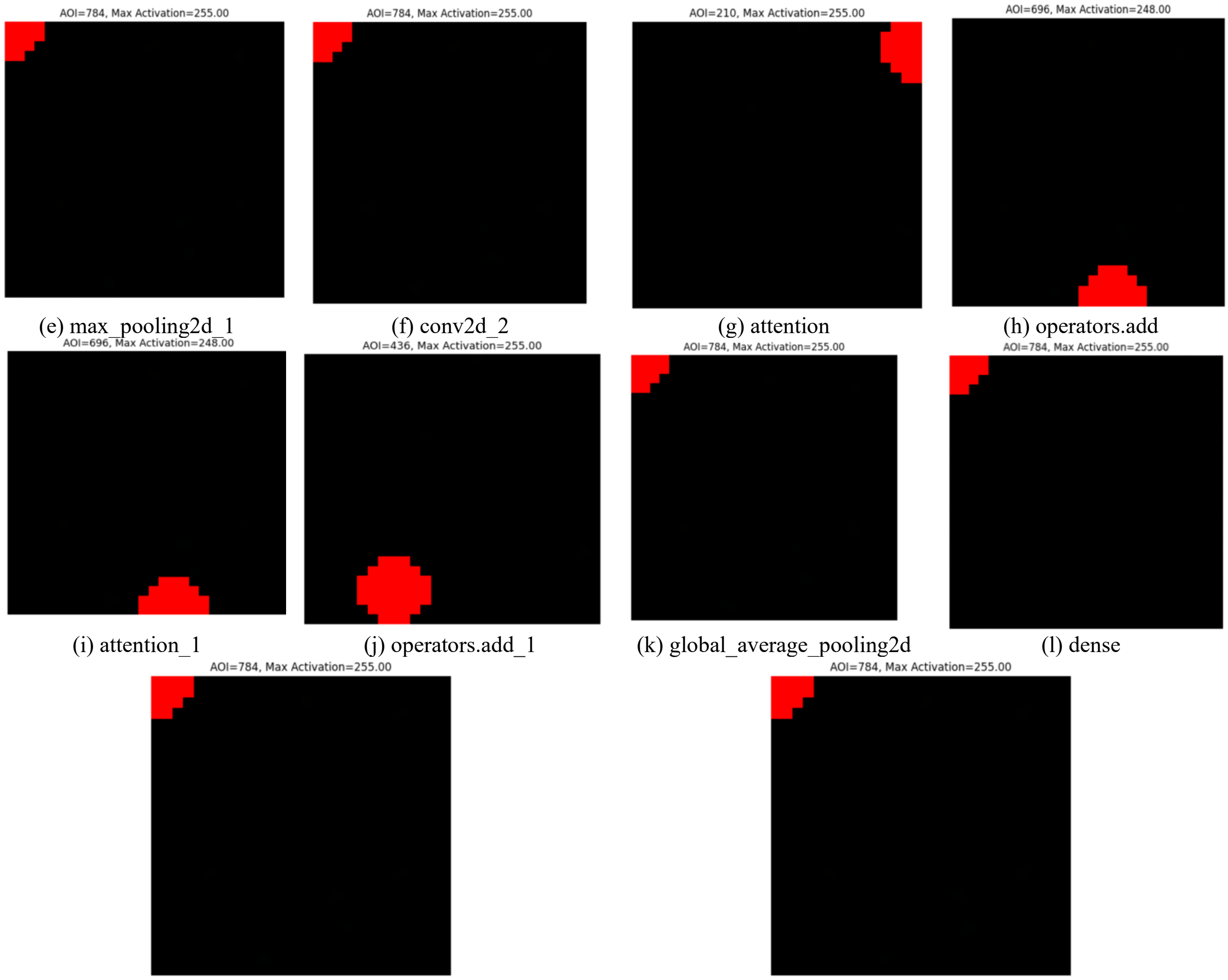
(b) conv2d



(c) max_pooling2d



(d) conv2d_1



(e) max_pooling2d_1 (f) conv2d_2 (g) attention (h) operators.add

(i) attention_1 (j) operators.add_1 (k) global_average_pooling2d (l) dense

(m) dropout (n) dense_1

Figure 4.58: Grad-CAM of each layer of LiteViT_APSO

Table 4.18: Layer-wise Grad-CAM Values for LiteViT_APSO

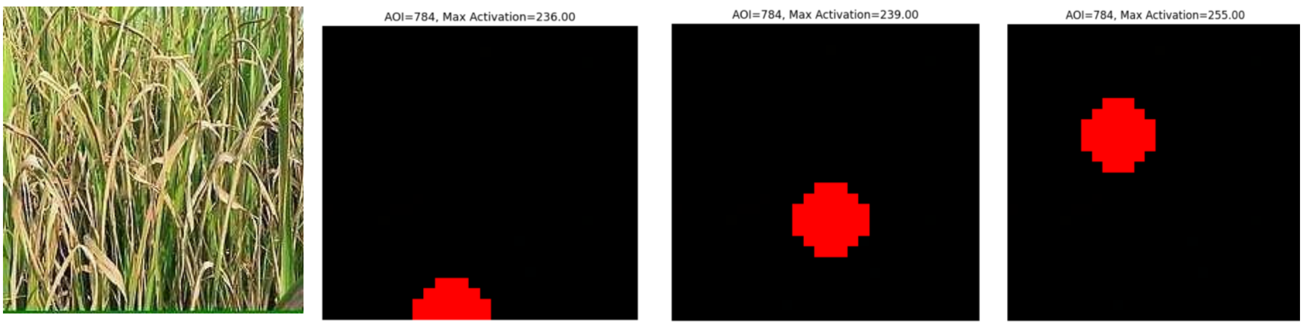
Layer	AOI	Max activation
conv2d	784	249
max_pooling2d	784	248
conv2d_1	784	255
max_pooling2d_1	784	255
conv2d_2	784	255
attention	210	255
operators.add	696	248
attention_1	696	248
operators.add_1	436	255
global_average_pooling2d	784	255

dense	784	255
dropout	784	255
dense_1	784	255

The Grad-CAM analysis for LiteViT_APSO is summarized in Table 4.18, detailing the AOI and maximum activation values for each layer. Beginning with the 'conv2d' layer, the model exhibits a substantial AOI of 784 pixels with a maximum activation of 249, indicating significant attention to specific regions. This pattern continues in subsequent layers such as 'max_pooling2d' and 'conv2d_1,' maintaining consistent AOI sizes and reaching the maximum activation limit of 255. However, the 'attention' layer marks a shift with a reduced AOI of 210 pixels while maintaining high activation. In contrast, 'operators.add' and 'attention_1' layers maintain larger AOIs but experience slight decreases in maximum activation. Finally, 'operators.add_1' showcases a reduced AOI yet retains maximum activation, emphasizing specific significant regions. The layers that follow, including 'global_average_pooling2d,' 'dense,' 'dropout,' and 'dense_1,' exhibit a consistent AOI size of 784 pixels, with the model assigning equal importance to these regions. The maximum activation values remain at 255, reaffirming the significance of these areas in the prediction process. The variations in AOI sizes and maximum activation values across layers are indicative of the LiteViT_APSO model's evolving focus as it processes input images. These changes demonstrate the model's adaptability in allocating attention to specific regions, resulting in increasingly refined decision-making as the image features are hierarchically analyzed. The shift in the red circle's position across layers signifies the evolving Areas of Interest, as the model fine-tunes its predictions based on the most critical features within the input image.

d) LiteViT_WWO

The Grad-CAM analysis of the LiteViT_WWO model, as depicted in Figure 4.59, offers a detailed examination of each layer's decision-making process in image classification. By applying Grad-CAM techniques to individual layers, the study aims to uncover the model's attention and feature activation patterns. A representative image of Brown Leaf Blast is used as input, and Grad-CAM visualizations are generated for each layer, highlighting Areas of Interest (AOIs) crucial for predictions. These AOIs, marked by red circles, signify regions of paramount importance in the model's decision-making process, with their size and location varying across layers.



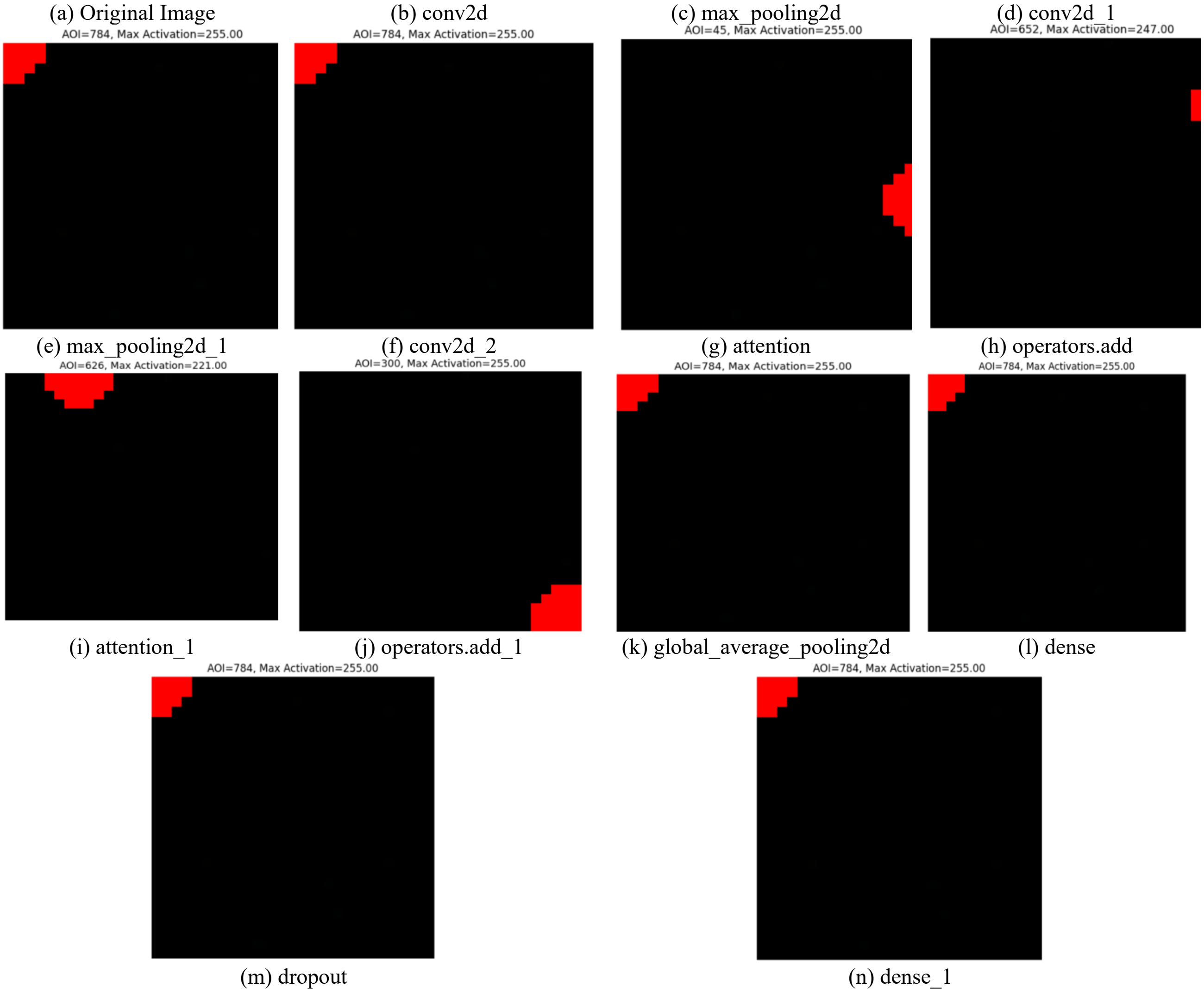


Figure 4.59: Grad-CAM of each layer of LiteViT_WWO

Table 4.19: Layer-wise Grad-CAM Values for LiteViT_WWO

Layer	**AOI**	**Max activation**
conv2d	784	236
max_pooling2d	784	239
conv2d_1	784	255
max_pooling2d_1	784	255
conv2d_2	784	255
attention	45	255
operators.add	652	247
attention_1	626	221
operators.add_1	300	255

global_average_pooling2d	784	255
dense	784	255
dropout	784	255
dense_1	784	255

Each Grad-CAM visualization includes a color intensity scale indicating the maximum activation value within the AOI, allowing researchers to discern the model's focus. A consistent heatmap threshold of 0.3 is applied to precisely identify AOI pixels, ensuring control over AOI size and granularity. Table 4.19 provides a detailed breakdown of the Grad-CAM analyses, layer by layer. Starting with Conv2D, a substantial AOI of 784 pixels with a maximum activation of 236 is observed, indicating the model's initial focus. Subsequent layers exhibit varying AOIs and maximum activations, reflecting the model's evolving attention and feature extraction strategy.

The presence of red circles in these Grad-CAM visualizations is crucial in pinpointing the AOI within each layer. These circles vary in size and location across layers, acting as a dynamic indicator of where the model attributes the highest importance in making its predictions. The consistent application of a heatmap threshold of 0.3 ensures that these AOIs are well-defined and enables meticulous control over their size and granularity. This thresholding approach offers a mechanism to regulate the model's attention, which is pivotal in understanding its decision-making process at different architectural levels. In summary, Table 14 exemplifies the dynamic and adaptive nature of LiteViT_WWO's hierarchical feature learning and decision-making process as it traverses through its distinct layers.

e) LiteViT_APSWWO

This study employs Grad-CAM techniques to dissect the LiteViT_APSWWO model's decision-making process for image classification. Figure 4.60 showcases Grad-CAM visualizations for each model layer, revealing critical Areas of Interest (AOIs) marked by bold red circles. These AOIs represent spatial regions crucial for predictions, with their size and location evolving across layers, indicating the model's progressive feature learning. Bright colors in the visualizations denote high model attention, while a consistent heatmap threshold ensures precise AOI identification.

Table 4.20 provides a detailed breakdown of LiteViT_APSWWO's layers, elucidating their decision-making dynamics via Grad-CAM analysis. Starting with 'conv2d,' a broad AOI of 784 and high Max Activation (251) signify the initial layer's comprehensive focus. Subsequent layers maintain broad AOIs while intensifying focus on specific regions, as seen in 'conv2d_1' with a Max Activation of 255. The 'attention' layer introduces focused AOIs, refining attention to 356 pixels while maintaining high Max Activation. 'Operators.add' widens focus with a broad AOI and slightly reduced Max Activation (252), indicating varied attention dynamics. 'Global_average_pooling2d' and 'dense' layers maintain broad AOIs and high Max Activation, emphasizing overall image importance. 'Dropout' and 'dense_1' layers echo this pattern, indicating consistent attention throughout the model's final stages. The varying AOIs and Max Activation values illustrate LiteViT_APSWWO's adaptive decision-making, crucial for accurate classifications. The evolving AOIs depicted in Grad-CAM images underscore LiteViT_APSWWO's dynamic attention allocation across layers, crucial for hierarchical feature learning.

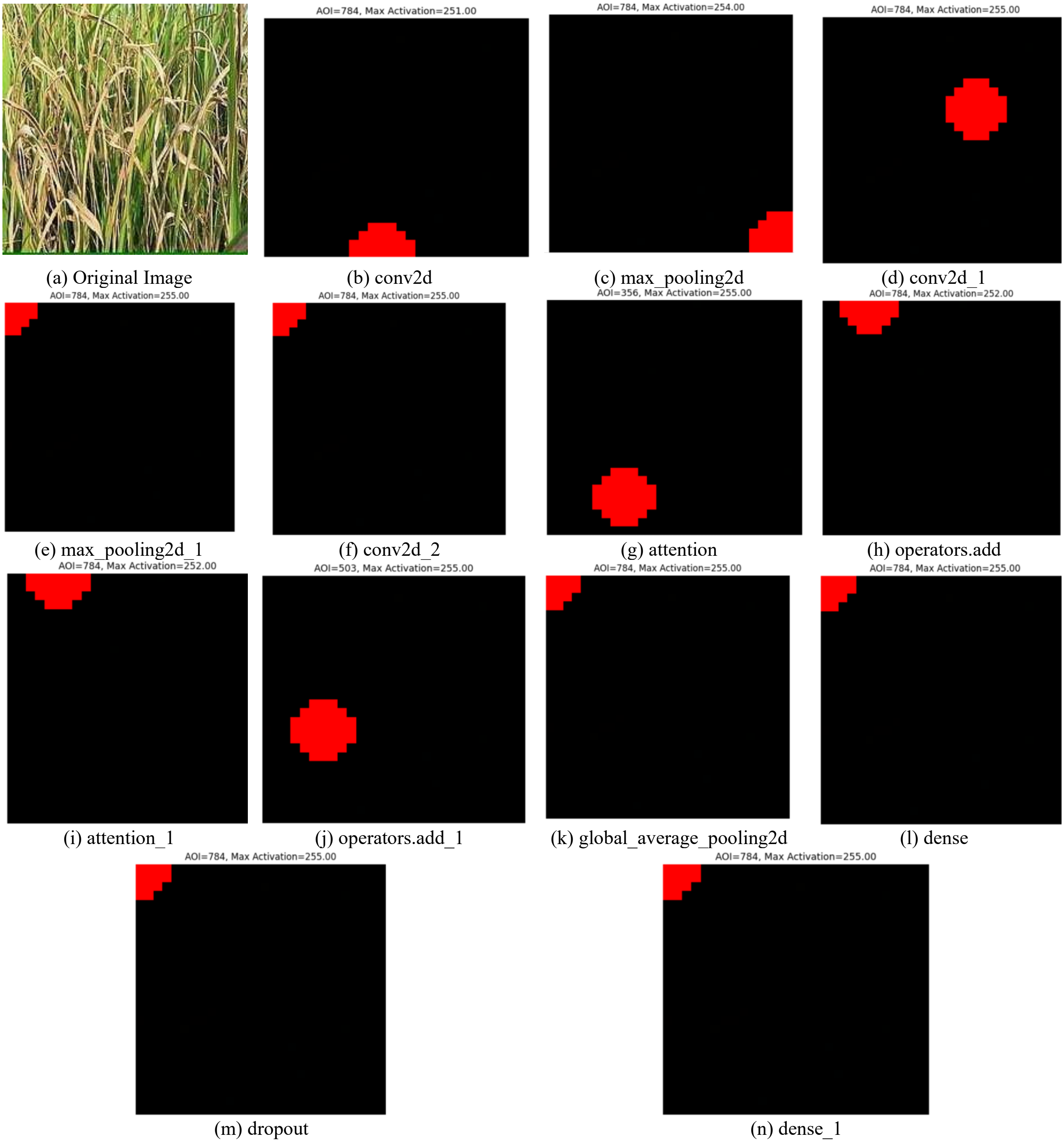


Figure 4.60: Grad-CAM of each layer of LiteViT_APSWWO

Table 4.20: Layer-wise Grad-CAM Values for LiteViT_APSWWO

Layer	AOI	Max activation
conv2d	784	251
max_pooling2d	784	254
conv2d_1	784	255
max_pooling2d_1	784	255
conv2d_2	784	255
attention	356	255
operators.add	784	252
attention_1	784	252
operators.add_1	503	255
global_average_pooling2d	784	255
dense	784	255
dropout	784	255
dense_1	784	255

4.4 Comparative Analysis

A comprehensive comparative analysis has been conducted between various CNN techniques, including CNN_Original, CNN_PSO, CNN_APSO, CNN_WWO, and CNN_APSWWO, and LiteViT techniques, encompassing LiteViT_Original, LiteViT_PSO, LiteViT_APSO, LiteViT_WWO, and LiteViT_APSWWO. The objective was to evaluate their effectiveness in detecting paddy diseases. Subsequently, the top-performing CNN and LiteViT models were compared to determine the most effective model for paddy disease detection.

4.4.1 Convolutional Neural Network

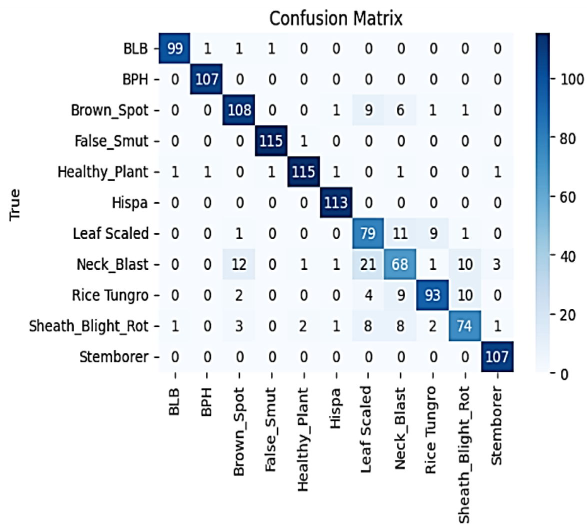
In this section, a comparative analysis is conducted among the CNN_Original, CNN_PSO, CNN_APSO, CNN_WWO, and CNN_APSWWO models in their classification of paddy diseases into categories including BLB, BPH, brown spot, false smut, healthy plant, hispa, leaf scaled, neck blast, tungro, sheath blight rot, and stemborer. This analysis aims to assess and contrast the performance of these different techniques.

Various CNN techniques, including CNN_Original, CNN_PSO, CNN_APSO, CNN_WWO, and CNN_APSWWO, are employed in this study as shown in Table 4.21, each with distinct hyperparameter configurations. CNN_Original utilizes a learning rate of 0.001, a batch size of 16, 100 epochs, and dropout rates of 0.5 for the CNN layers, 0.2 for the Attention layer, and 0.5 for Attention_1. In contrast, CNN_PSO employs a learning rate of 0.004, a batch size of 147, 64 epochs, and dropout rates of 0.4, 0.4, and 0.5 for the CNN, Attention, and

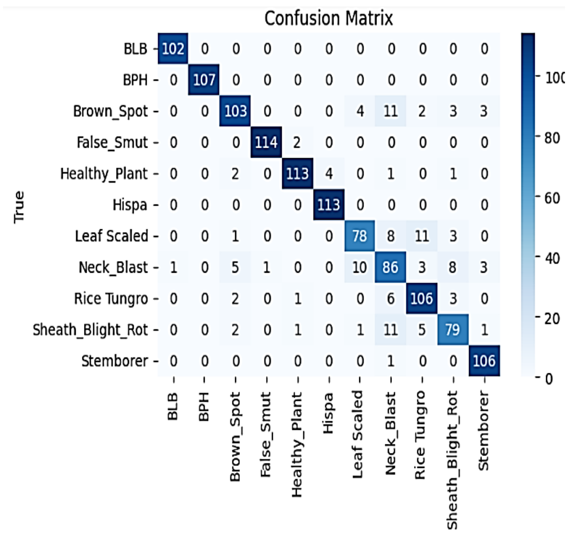
Attention_1 layers, respectively. CNN_APSO utilizes a learning rate of 0.001, a batch size of 168, 99 epochs, and dropout rates of 0.5, 0.2, and 0.5 for the corresponding layers. CNN_WWO opts for a learning rate of 0.002, a batch size of 114, 77 epochs, and dropout rates of 0.6, 0.4, and 0.5 for the CNN, Attention, and Attention_1 layers. Lastly, CNN_APSWWO sets a learning rate of 0.001, a batch size of 128, 176 epochs, and dropout rates of 0.5, 0.5, and 0.4 for the respective layers. These hyperparameters are instrumental in configuring and training each CNN technique, influencing their performance in paddy disease classification tasks.

Table 4.21: Optimized values of CNN

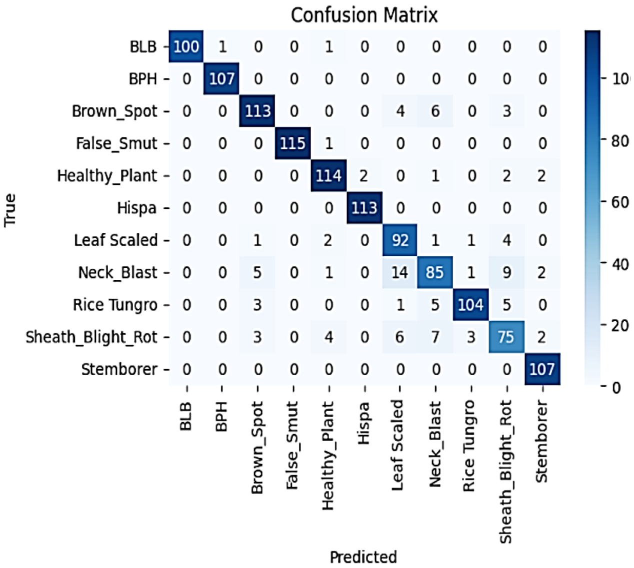
CNN Techniques	**Learning rate**	**Batch size**	**Epoch number**	**Dropout rate for CNN**	**Dropout rate for Attention layer**	**Dropout rate for Attention_1 layer**
CNN_Original	0.001	16	100	0.5	0.2	0.5
CNN_PSO	0.004	147	64	0.4	0.4	0.5
CNN_APSO	0.001	168	99	0.5	0.2	0.5
CNN_WWO	0.002	114	77	0.6	0.4	0.5
CNN_APSWWO	0.001	128	176	0.5	0.5	0.4



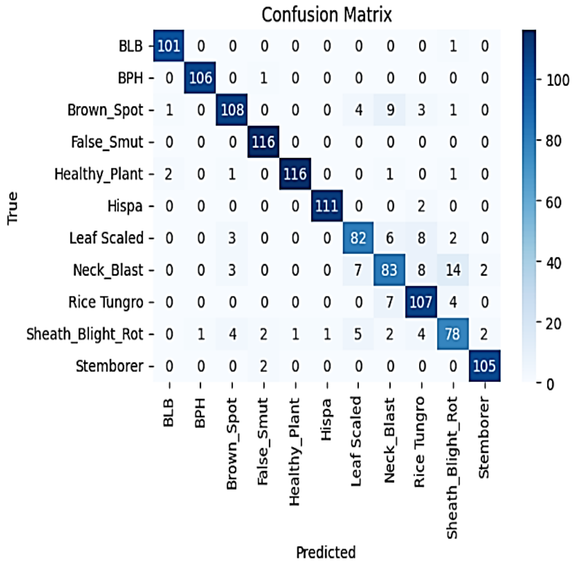
(a) CNN_Original



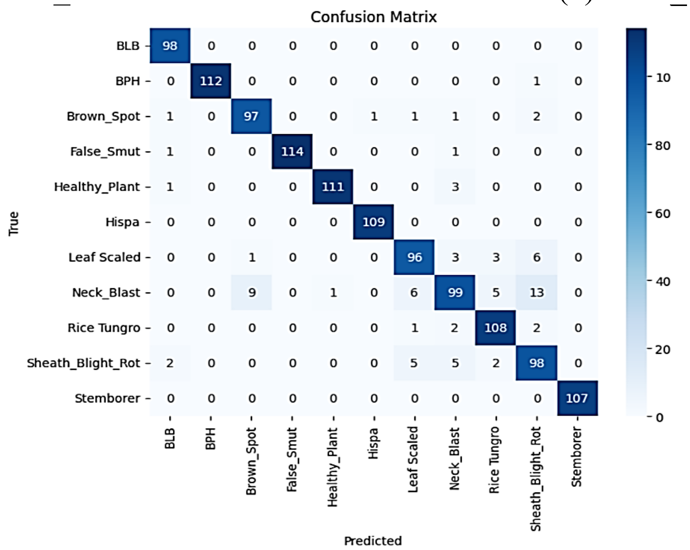
(b) CNN_PSO



(c) CNN_APSO



(d) CNN_WWO

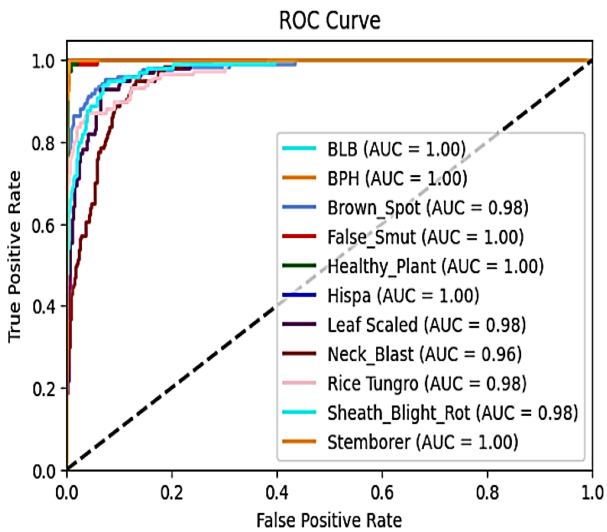


(e) CNN_APSWWO

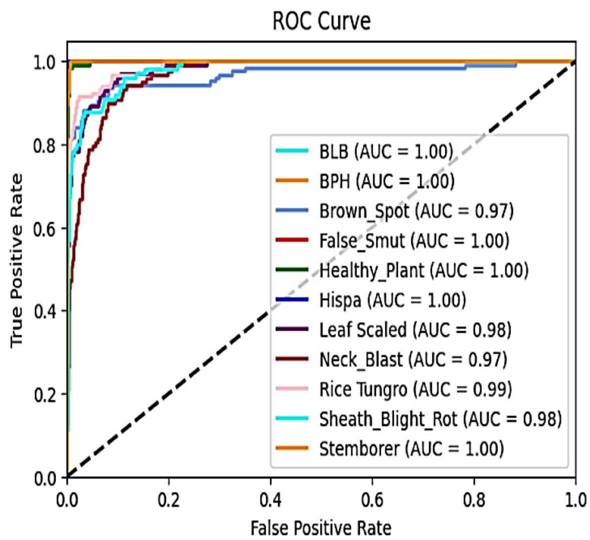
Figure 4.61: Confusion matrix of CNN

Figure 4.61, Figure 4.62, Figure 4.63, and Figure 4.64 represent the confusion matrix, ROC curve, precision-recall curve, and learning curves of CNN_Original, CNN_PSO, CNN_APSO, CNN_WWO, and CNN_APSWWO. Table 4.22 represents the performance metrics of these techniques. CNN_Original achieves an average of 98 correctly classified instances, 19.82 average misclassified instances, and scores 87.73% in recall, 87.82% in precision, and 87.73% in F1-score. It exhibits a training loss of 0.1884, a testing loss of 1.103, a ROC-AUC score of 0.983, an AP value of 0.931, an error rate of 12%, and an accuracy of 88%. CNN_PSO records an average of 100.64 correctly classified instances, 11 average misclassified instances, with recall at 90.27%, precision at 90.18%, and F1-score at 90.09%. The model shows a training loss of 0.0766, a testing loss of 0.6784, a ROC-AUC score of 0.99, an AP value of 0.939, an error rate of 10%, and an accuracy of 90%. CNN_APSO demonstrates an average of 102.27 correctly classified instances, 9.36 average misclassified instances, recall at 91.55%, precision at 91.64%, and F1-score at 91.55%. Its training loss is 0.023, testing loss is 0.4936, ROC-AUC score is 0.994, AP value is 0.96, error rate is 8%, and accuracy is 92%. CNN_WWO displays an average of 101.18 correctly classified instances, 10.45 average misclassified instances, recall at 91.55%, precision at 90.64%, and F1-score at 90.55%. The training loss is 0.0457, testing loss is 0.5704, ROC-AUC score is 0.994, AP value is 0.955, error rate is 9%, and accuracy is 91%. CNN_APSWWO leads with an average of 104.45 correctly classified instances, 7.18 average misclassified instances, recall at 93.82%, precision at 94.09%, and F1-score at 93.91%. It maintains a training loss of 0.0019, testing loss of 0.4116, ROC-AUC score of 0.996, AP value of 0.975, error rate of 5%, and accuracy of 95%.

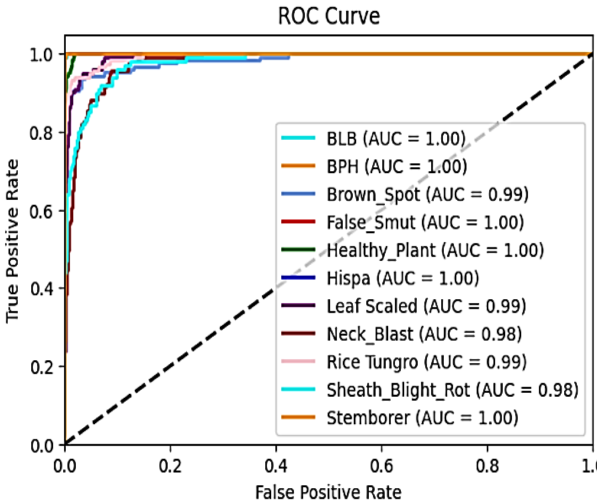
Figure 4.65 represents the graphical representation of performance metrics of CNN_Original, CNN_PSO, CNN_APSO, CNN_WWO, and CNN_APSWWO. In terms of correctly classified instances, CNN_APSWWO outperforms the others with an average of 104.45 instances, while CNN_Original ranks lowest with 98. Across these models, recall percentages range from 87.73% to 93.82%, with CNN_APSWWO achieving the highest recall. Precision percentages also demonstrate similar trends, with CNN_APSWWO leading at 94.09%. The F1-score, which combines precision and recall, places CNN_APSWWO in the top position at 93.91%. Training loss varies from 0.0019 to 0.1884, while testing loss ranges from 0.4116 to 1.103. ROC-AUC scores, indicating model discrimination, range from 0.983 to 0.996, with CNN_APSWWO achieving the highest score. Similarly, average precision (AP) values show CNN_APSWWO as the leader at 0.975. Error rates range from 5% to 12%, with CNN_APSWWO having the lowest error rate at 5%. Finally, accuracy percentages span from 88% to 95%, with CNN_APSWWO boasting the highest accuracy at 95%. Comparing these techniques, CNN_APSWWO consistently performs as the top performer across multiple metrics, with CNN_APSO closely following. CNN_WWO stands at the third position, while CNN_PSO ranks as the fourth. The CNN_Original ranks lower in terms of performance metrics.



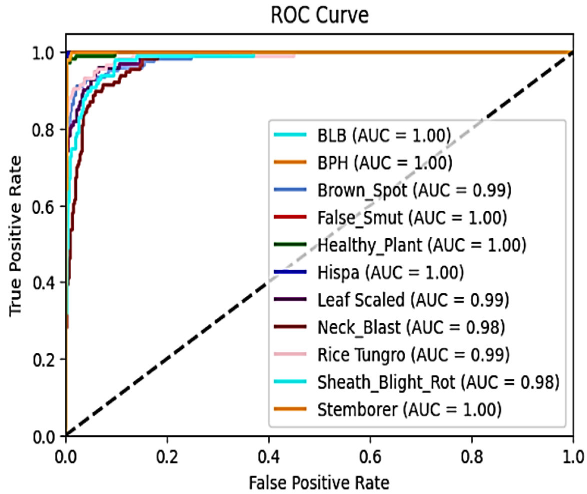
(a) CNN_Original



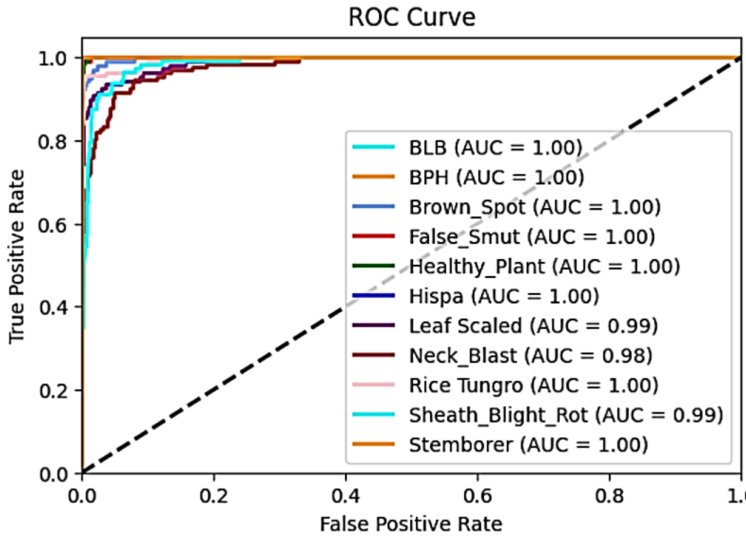
(b) CNN_PSO



(c) CNN_APSO

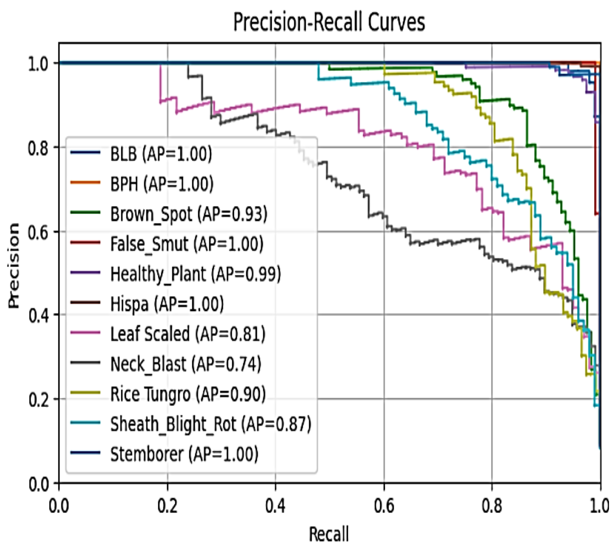


(d) CNN_WWO

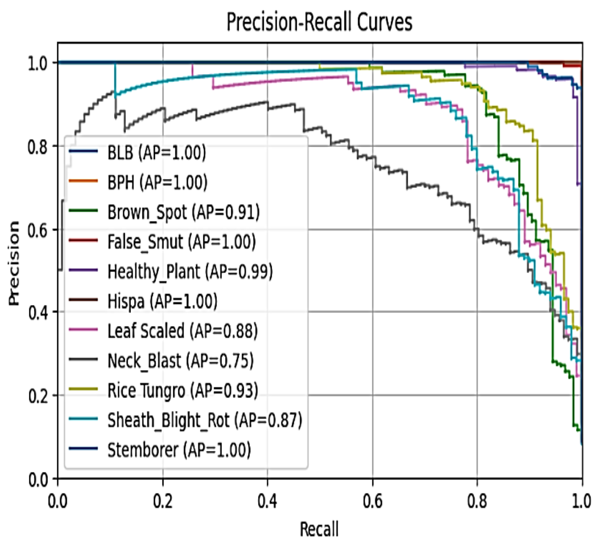


(e) CNN_APSWWO

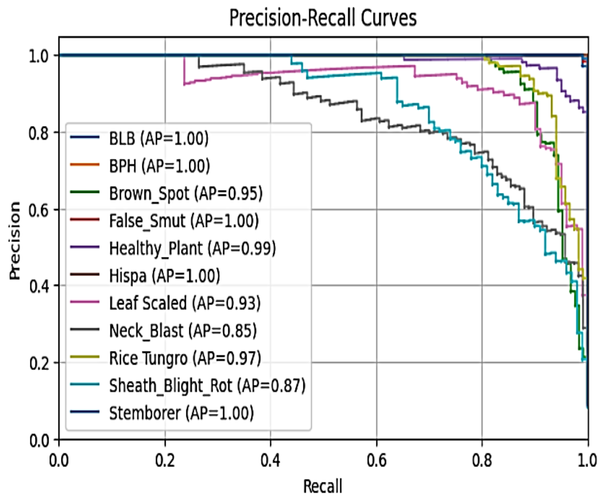
Figure 4.62: ROC curve of CNN



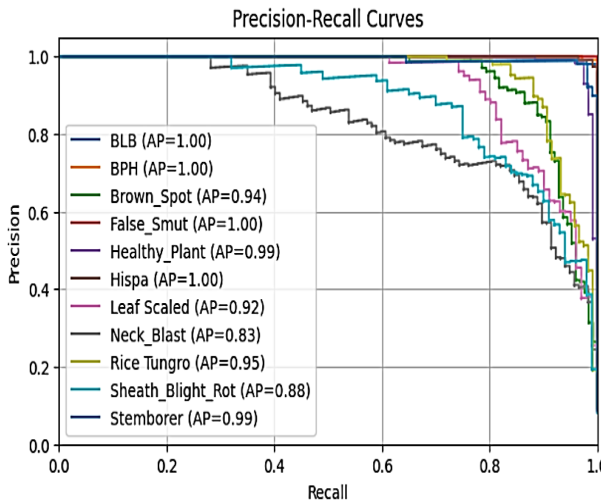
(a) CNN_Original



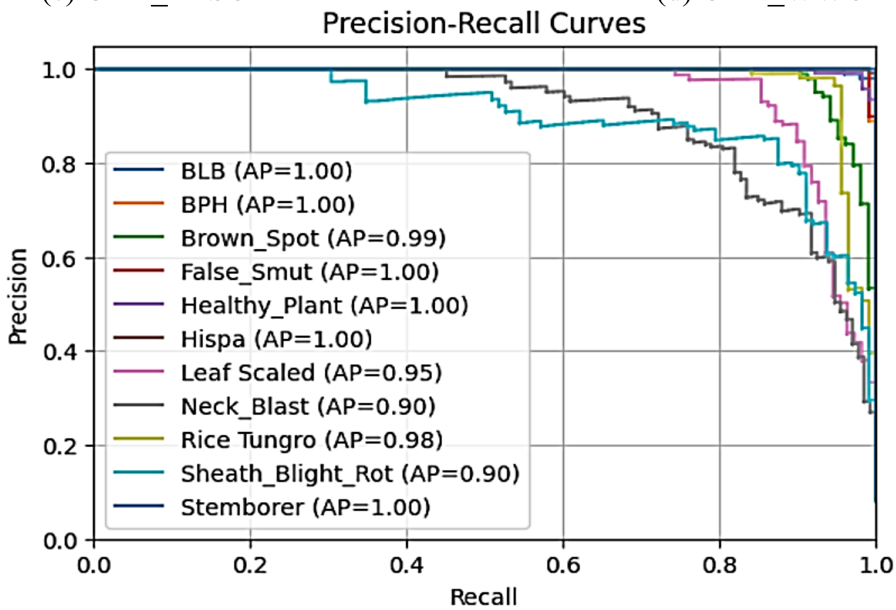
(b) CNN_PSO



(c) CNN_APSO

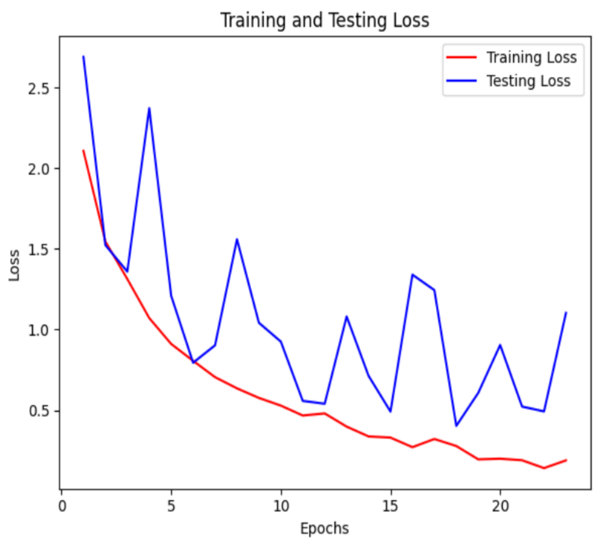


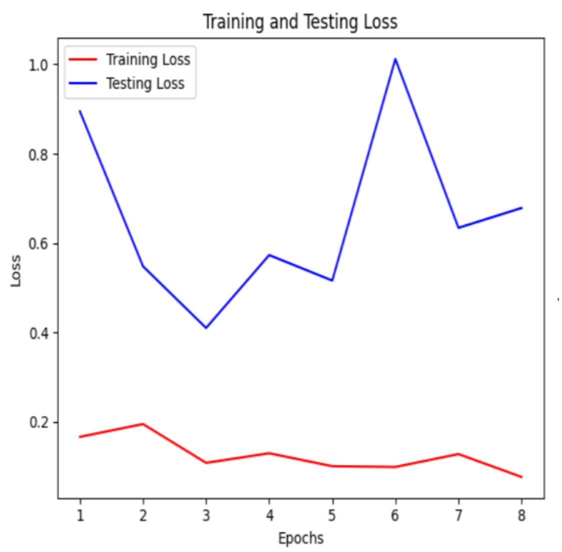
(d) CNN_WWO



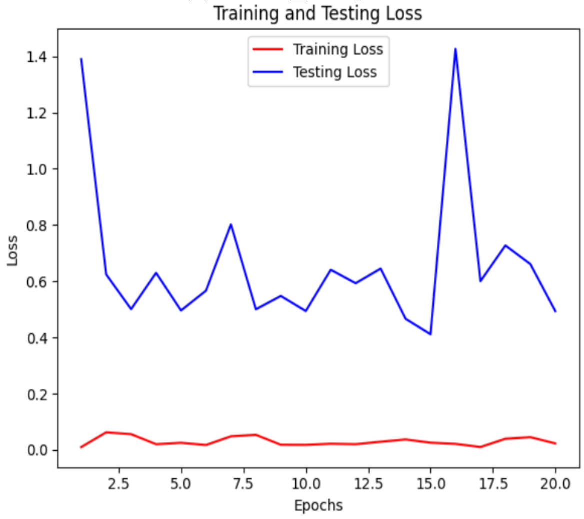
(e) CNN_APSWWO

Figure 4.63: Precision-recall curve of CNN

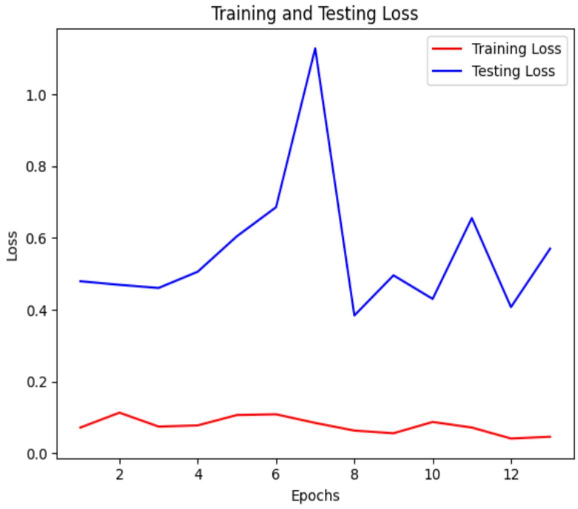




(a) CNN_Original



(b) CNN_PSO



(c) CNN_APSO (d) CNN_WWO

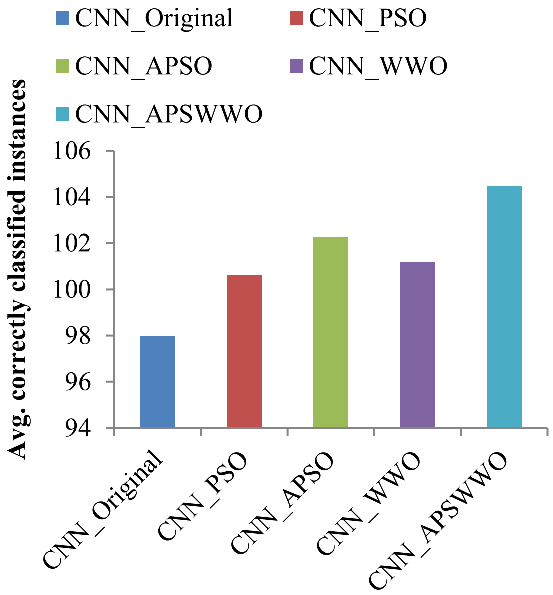
(e) CNN_APSWWO

Figure 4.64: Learning curve of CNN

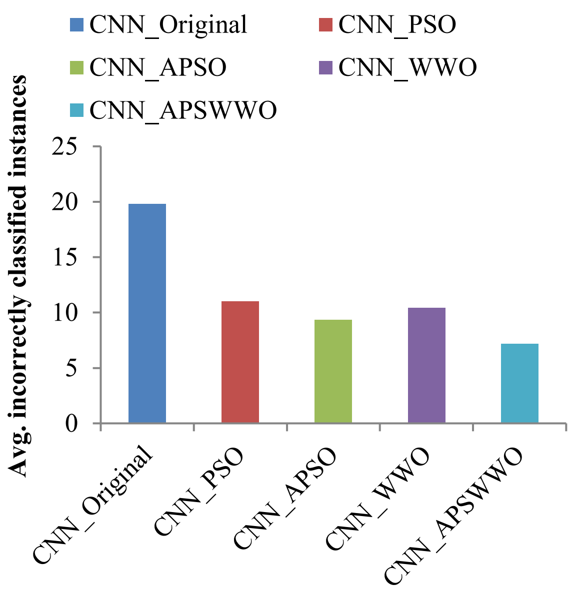
Table 4.22: Performance metrics of CNN

Performance Metrics	CNN_Original	CNN_PSO	CNN_APSO	CNN_WWO	CNN_APSWWO
Avg. correctly classified instances	98	100.64	102.27	101.18	104.45
Avg. incorrectly classified instances	19.82	11	9.36	10.45	7.18
Recall (%)	87.73	90.27	91.55	91.55	93.82
Precision (%)	87.82	90.18	91.64	90.64	94.09
F1-score (%)	87.73	90.09	91.55	90.55	93.91
Training loss	0.1884	0.0766	0.023	0.0457	0.0019
Testing loss	1.103	0.6784	0.4936	0.5704	0.4116

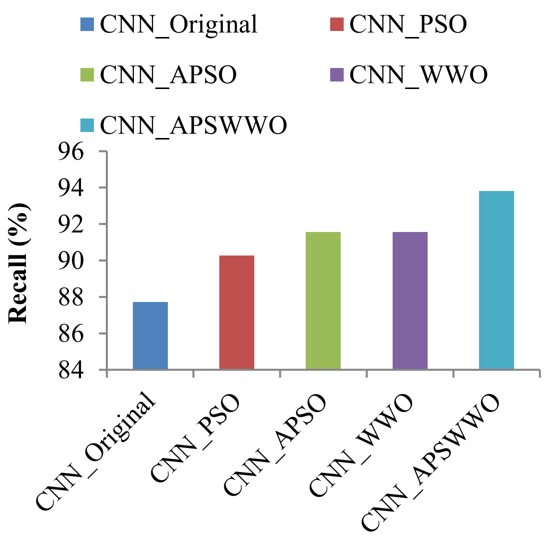
ROC-AUC score	0.983	0.99	0.994	0.994	0.996
AP value	0.931	0.939	0.96	0.955	0.975
Error rate (%)	12	10	8	9	5
Accuracy (%)	88	90	92	91	95



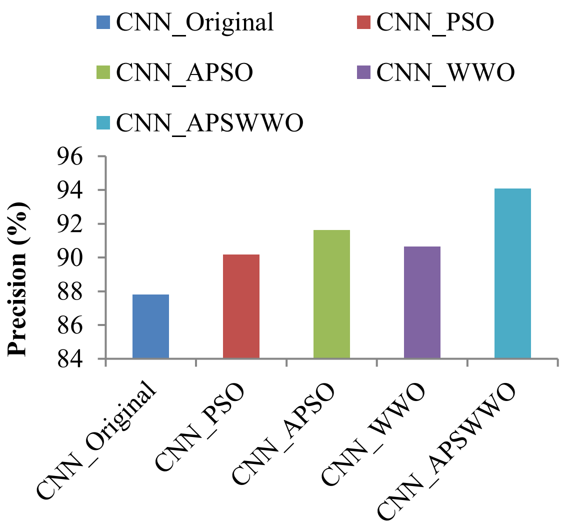
(a) Avg. correctly classified instances



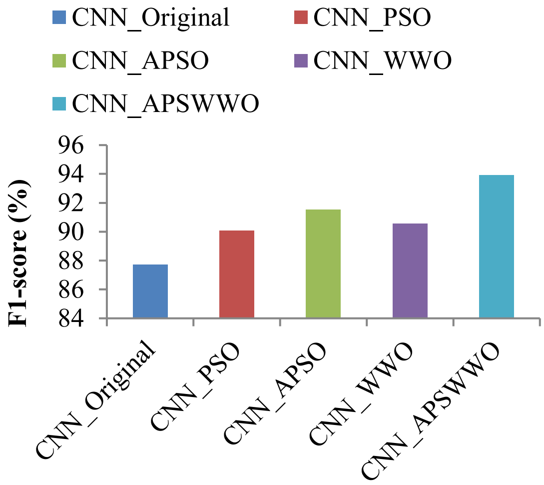
(b) Avg. incorrectly classified instances



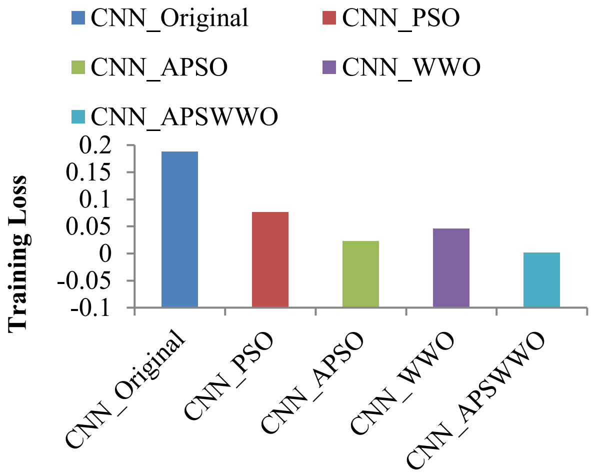
(c) Recall



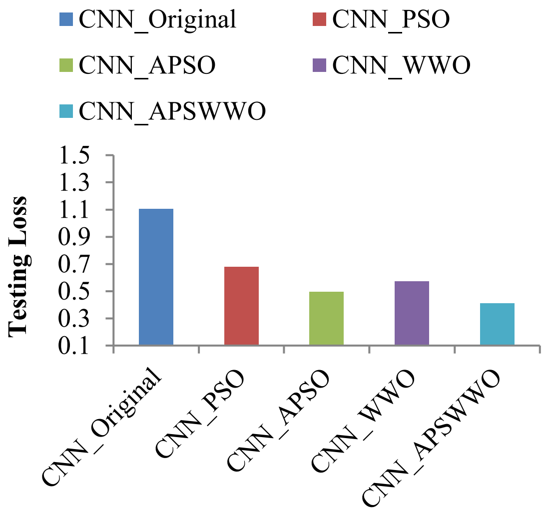
(d) Precision



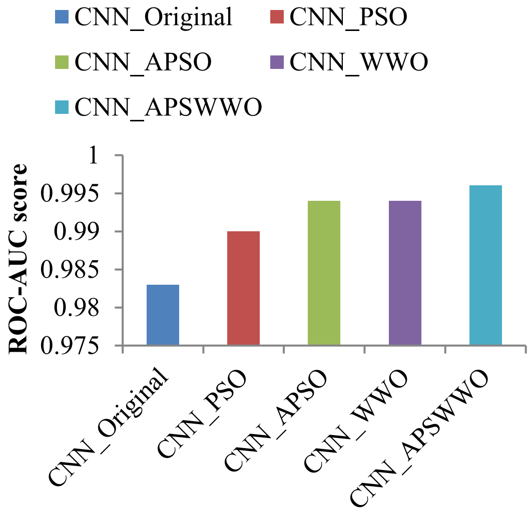
(e) F1-score



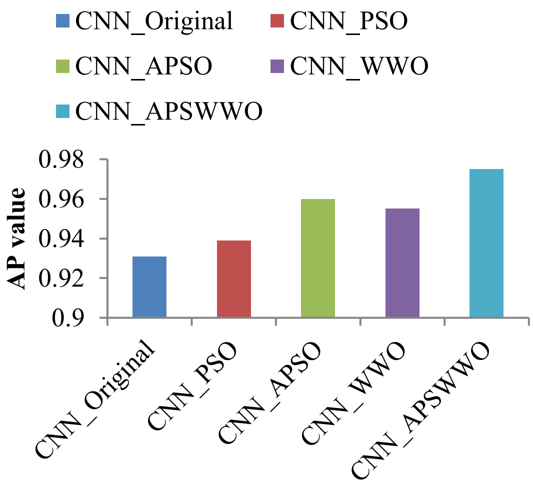
(f) Training Loss



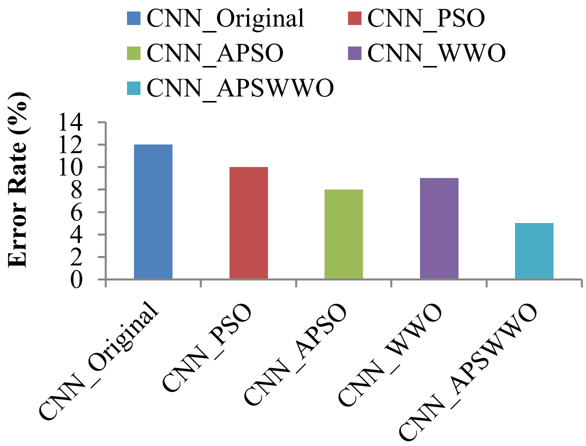
(g) Testing Loss



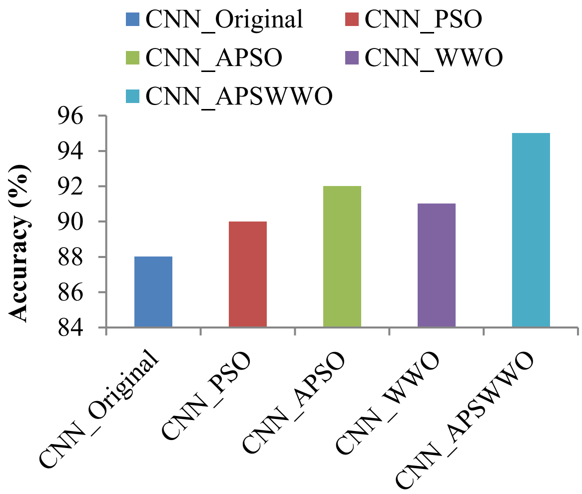
(h) ROC-AUC score



(i) Average Precision value



(j) Error Rate

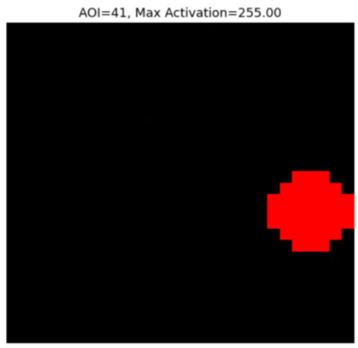


(k) Accuracy

Figure 4.65: Performance metrics of CNN

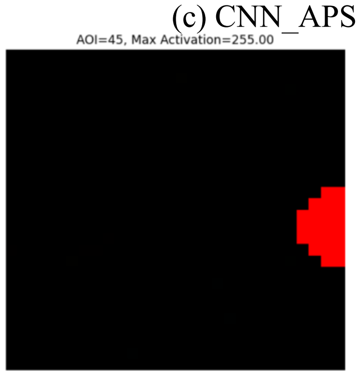
(a) CNN_Original

(b) CNN_PSO



(c) CNN_APSO

(d) CNN_WWO

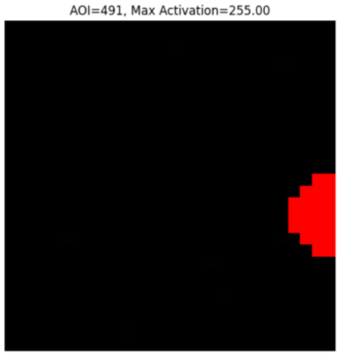


(e) CNN_APSWWO

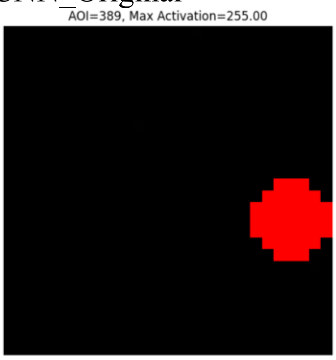
(i) Grad-CAM of conv2d layer of CNN

(a) CNN_Original

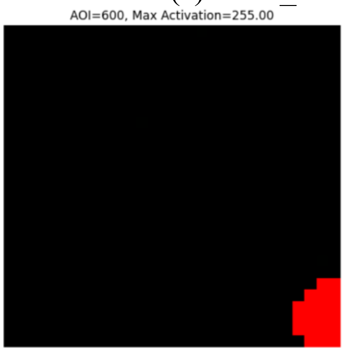
(b) CNN_PSO



(c) CNN_APSO

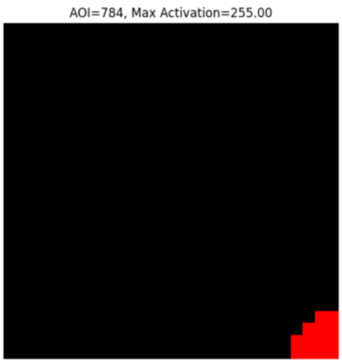


(d) CNN_WWO

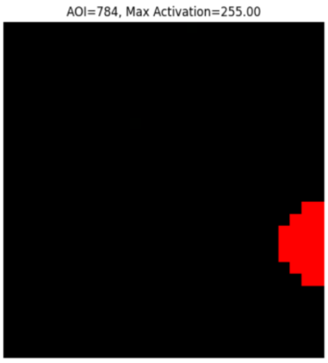


(e) CNN_APSWWO

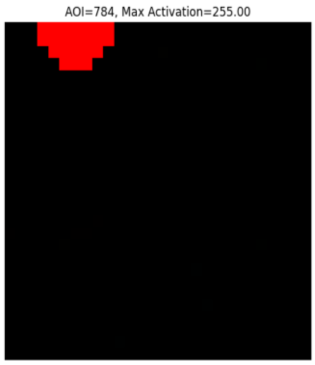
(ii) Grad-CAM of batch_normalization layer of CNN



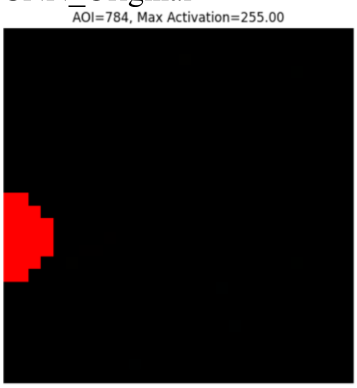
(a) CNN_Original



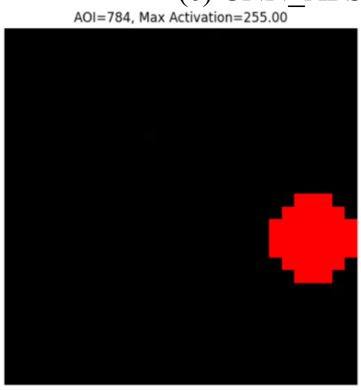
(b) CNN_PSO



(c) CNN_APSO

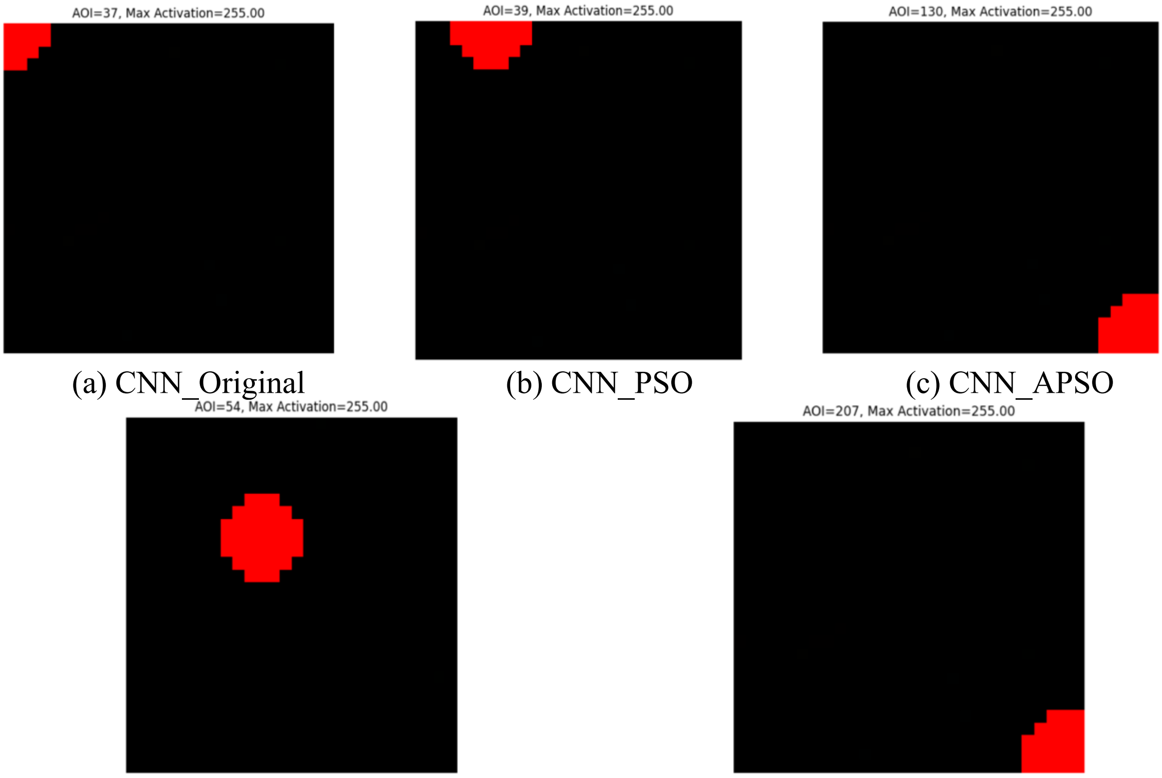


(d) CNN_WWO



(e) CNN_APSWWO

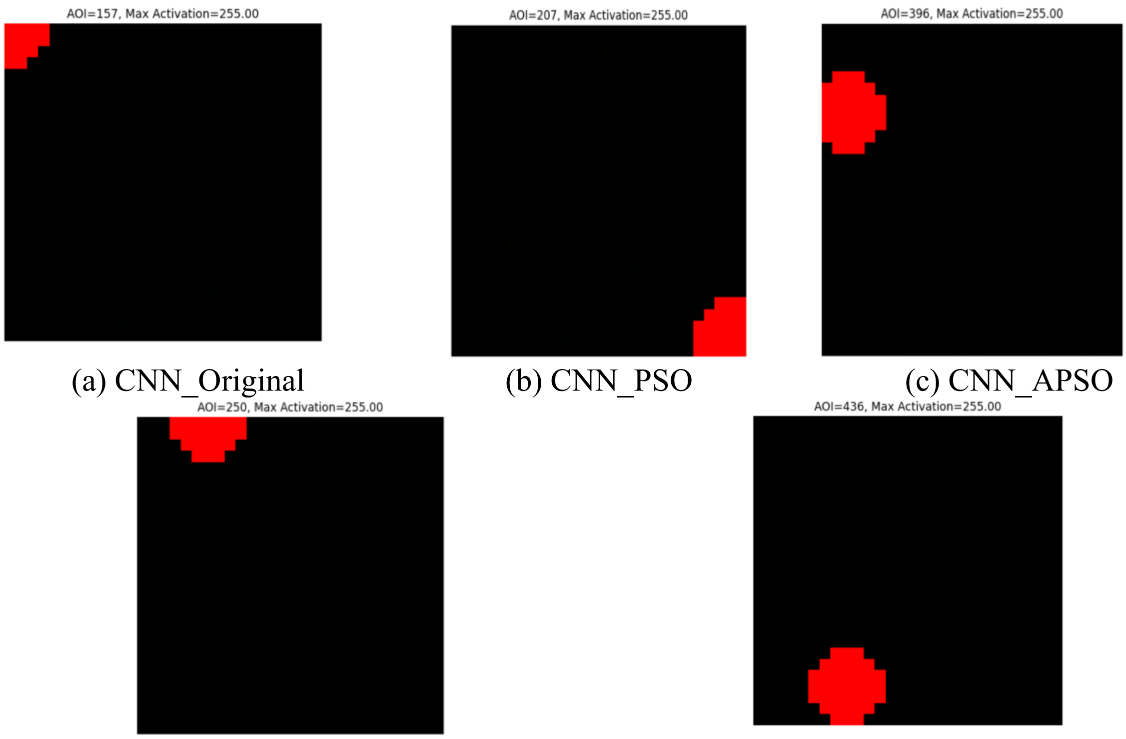
(iii) Grad-CAM of relu layer of CNN



(a) CNN_Original (b) CNN_PSO (c) CNN_APSO

(d) CNN_WWO (e) CNN_APSWWO

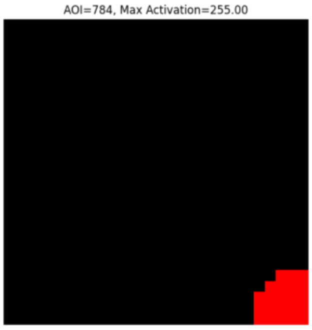
(iv) Grad-CAM of con2d_1 layer of CNN



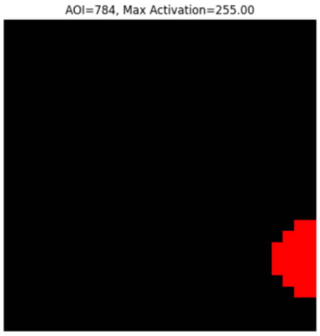
(a) CNN_Original (b) CNN_PSO (c) CNN_APSO

(d) CNN_WWO (e) CNN_APSWWO

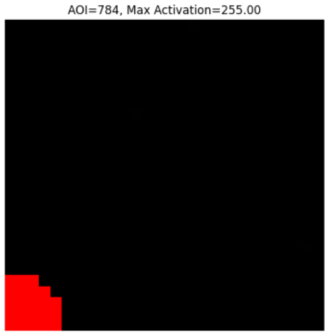
(v) Grad-CAM of batch_normalization_1 layer of CNN



(a) CNN_Original

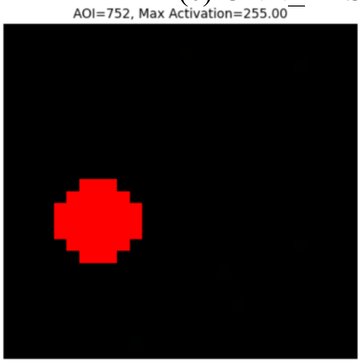


(b) CNN_PSO



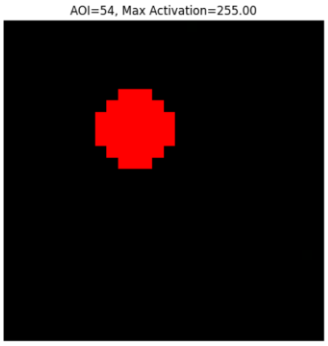
(c) CNN_APSO

(d) CNN_WWO

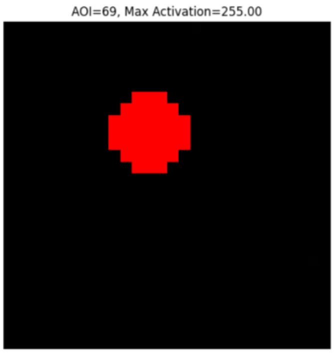


(e) CNN_APSWWO

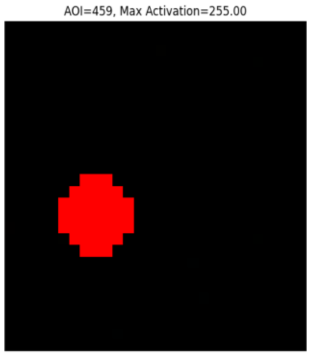
(vi) Grad-CAM of relu_1 layer of CNN



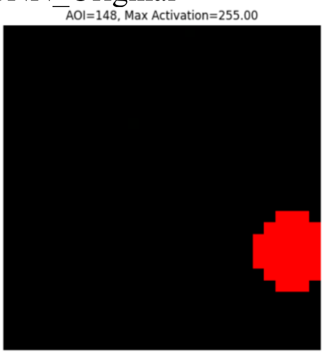
(a) CNN_Original



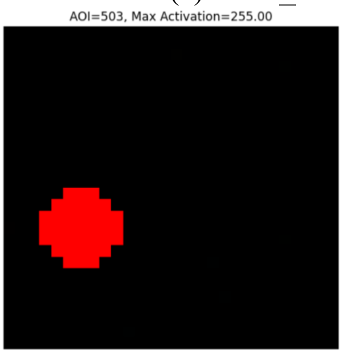
(b) CNN_PSO



(c) CNN_APSO

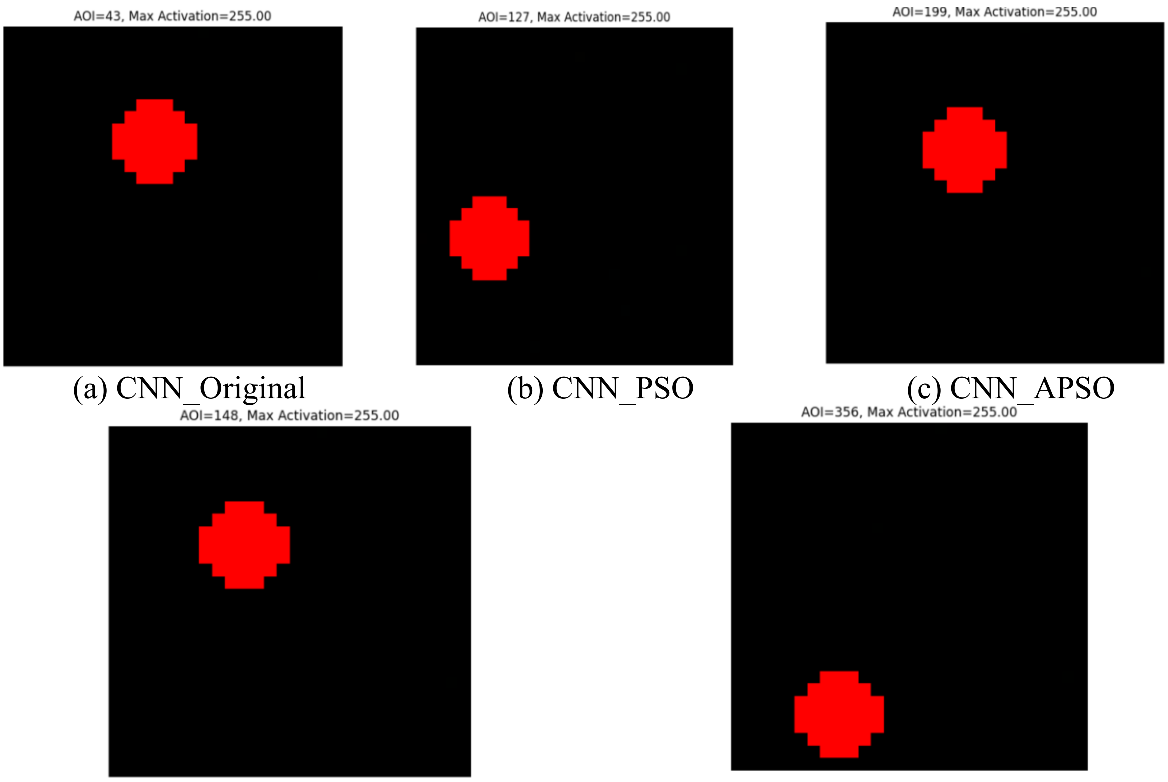


(d) CNN_WWO



(e) CNN_APSWWO

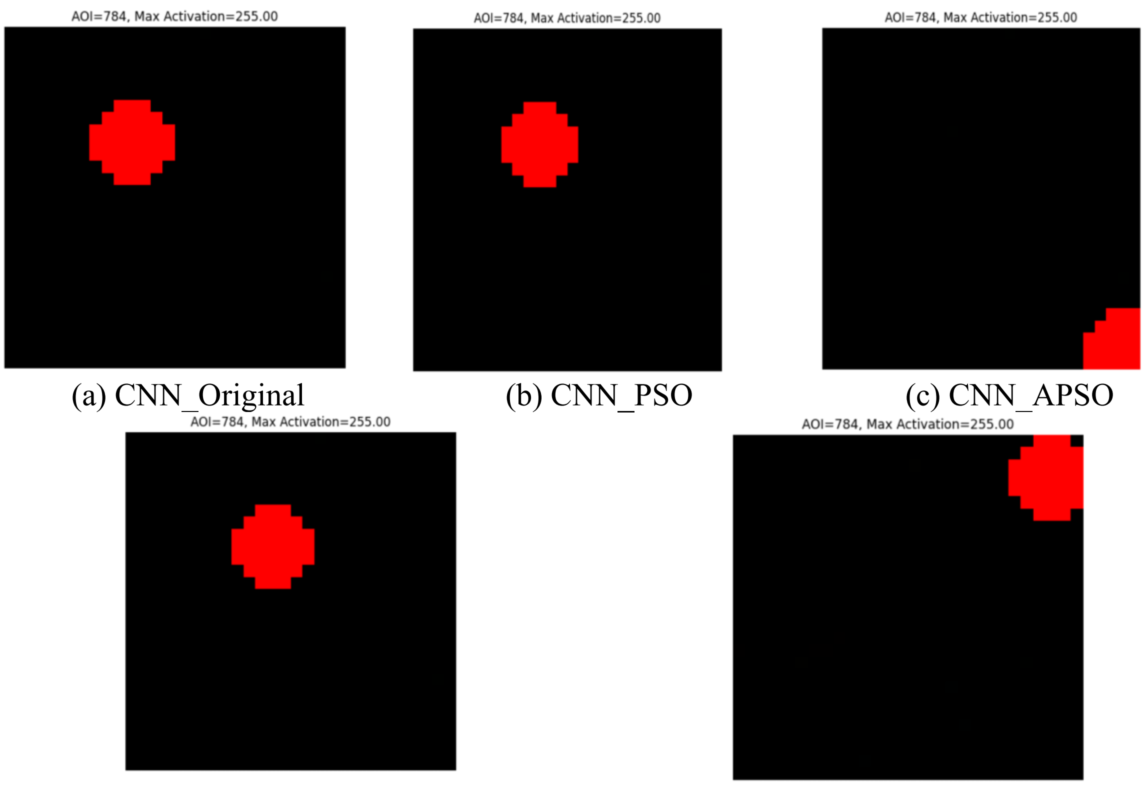
(vii) Grad-CAM of conv2d_2 layer of CNN



(a) CNN_Original (b) CNN_PSO (c) CNN_APSO

(d) CNN_WWO (e) CNN_APSWWO

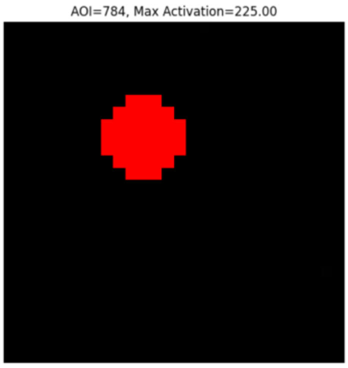
(viii) Grad-CAM of batch_normalization_2 layer of CNN



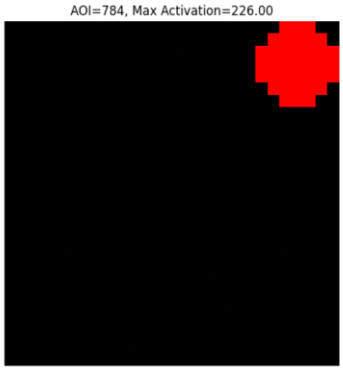
(a) CNN_Original (b) CNN_PSO (c) CNN_APSO

(d) CNN_WWO (e) CNN_APSWWO

(ix) Grad-CAM of relu_2 layer of CNN

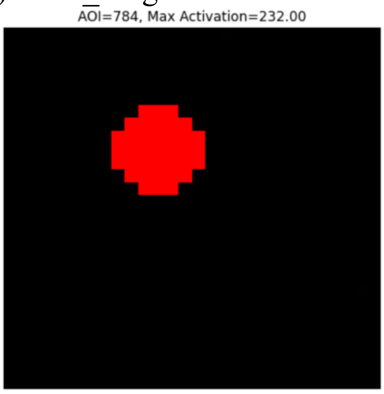


(a) CNN_Original

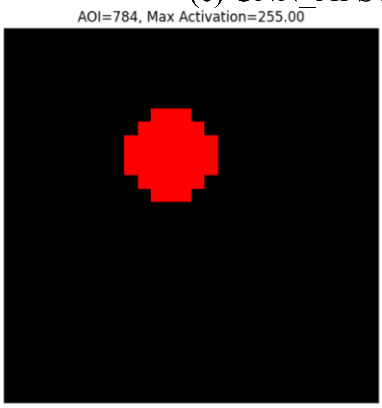


(b) CNN_PSO

(c) CNN_APSO

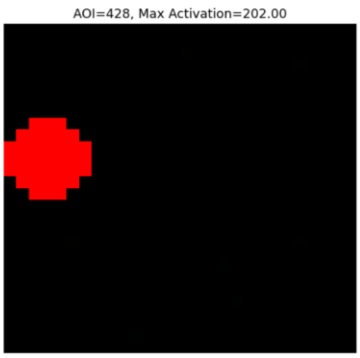


(d) CNN_WWO

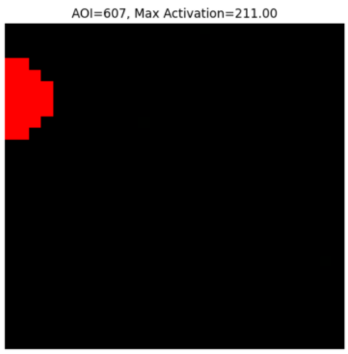


(e) CNN_APSWWO

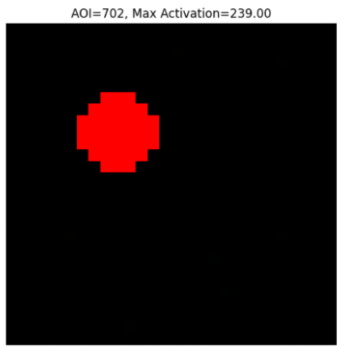
(x) Grad-CAM of max_pooling2d layer of CNN



(a) CNN_Original

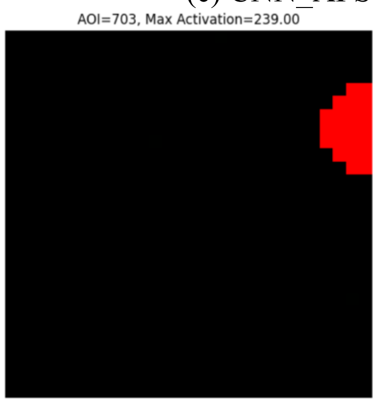


(b) CNN_PSO



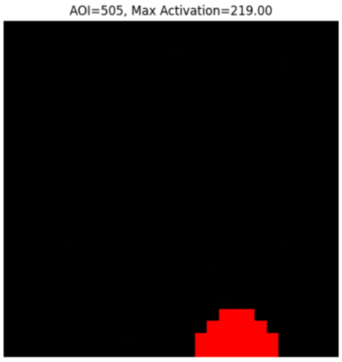
(c) CNN_APSO

(d) CNN_WWO

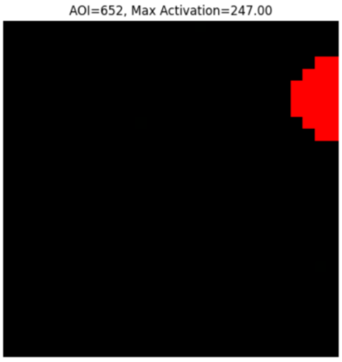


(e) CNN_APSWWO

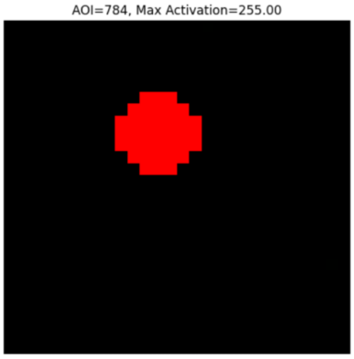
(xi) Grad-CAM of conv2d_3 layer of CNN



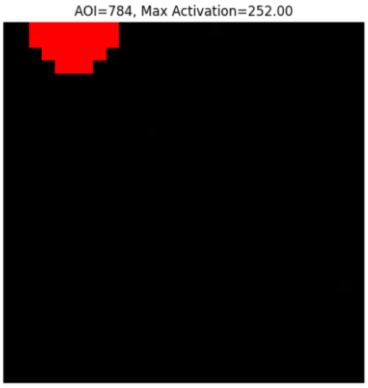
(a) CNN_Original



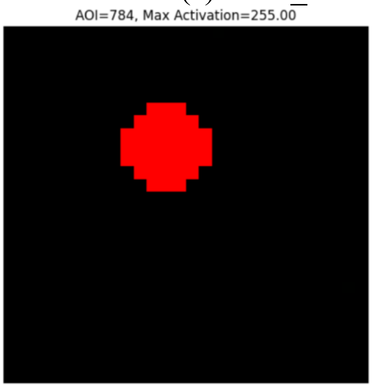
(b) CNN_PSO



(c) CNN_APSO



(d) CNN_WWO

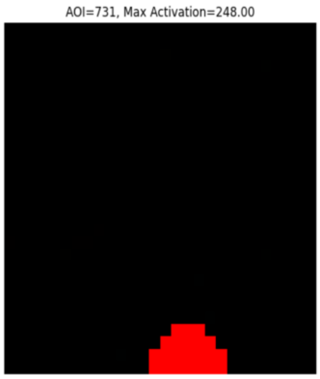


(e) CNN_APSWWO

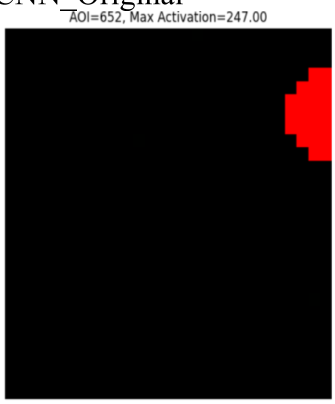
(xii) Grad-CAM of batch_normalization_3 layer of CNN

(a) CNN_Original

(b) CNN_PSO



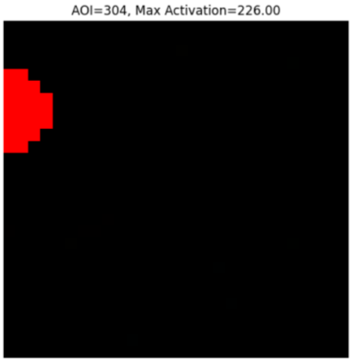
(c) CNN_APSO



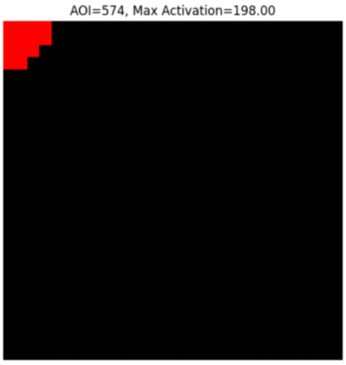
(d) CNN_WWO

(e) CNN_APSWWO

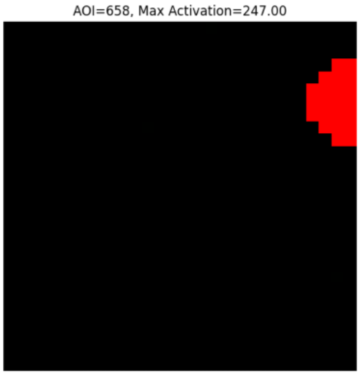
(xiii) Grad-CAM of math.multifly layer of CNN



(a) CNN_Original

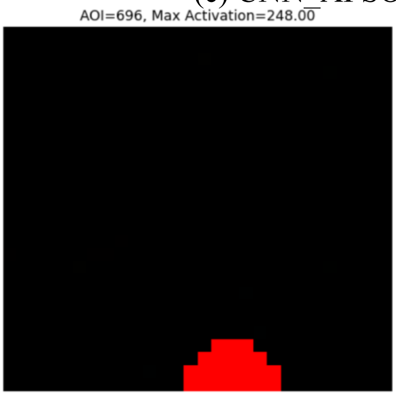


(b) CNN_PSO



(c) CNN_APSO

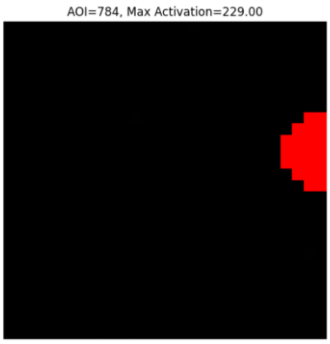
(d) CNN_WWO



(e) CNN_APSWWO

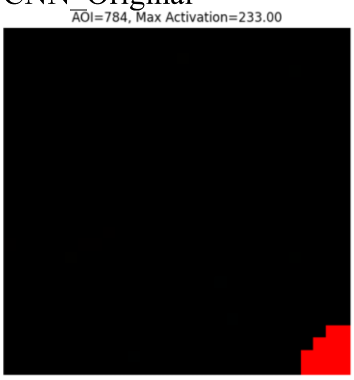
(xiv) Grad-CAM of attention layer of CNN

(a) CNN_Original

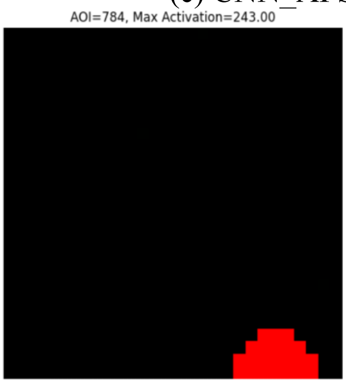


(b) CNN_PSO

(c) CNN_APSO

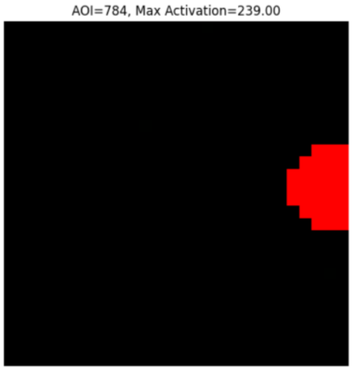


(d) CNN_WWO



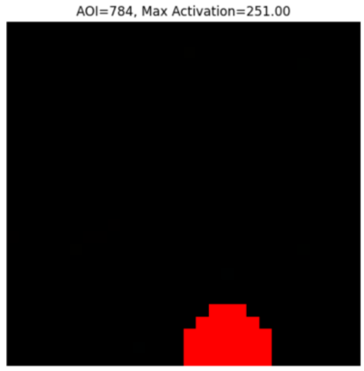
(e) CNN_APSWWO

(xv) Grad-CAM of math.multifly_1 layer of CNN



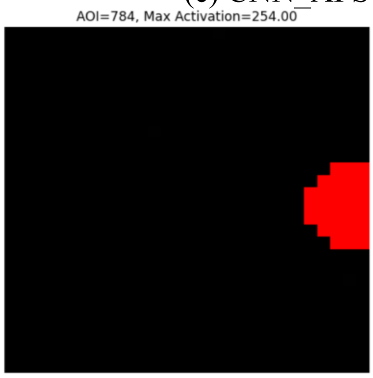
(a) CNN_Original

(b) CNN_PSO



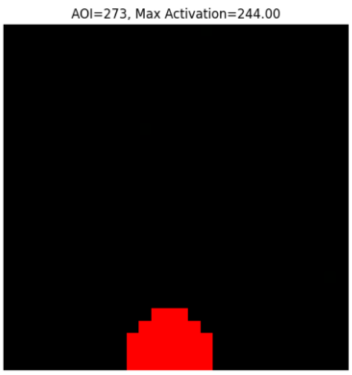
(c) CNN_APSO

(d) CNN_WWO

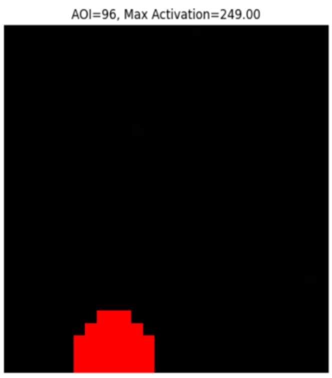


(e) CNN_APSWWO

(xvi) Grad-CAM of max_pooling2d_1 layer of CNN

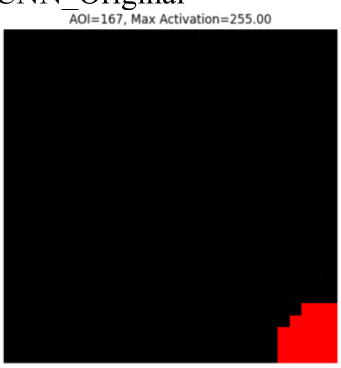


(a) CNN_Original

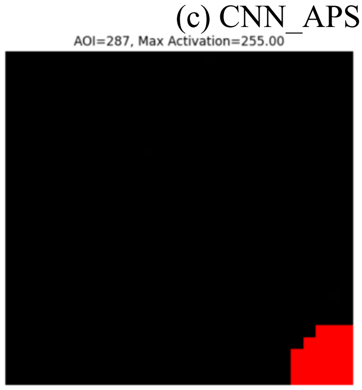


(b) CNN_PSO

(c) CNN_APSO

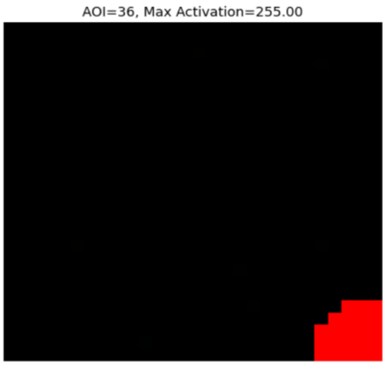


(d) CNN_WWO

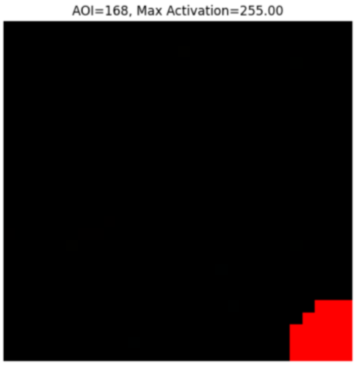


(e) CNN_APSWWO

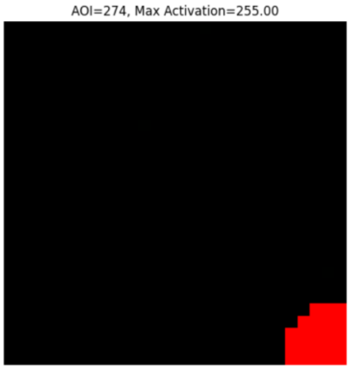
(xvii) Grad-CAM of conv2d_4 layer of CNN



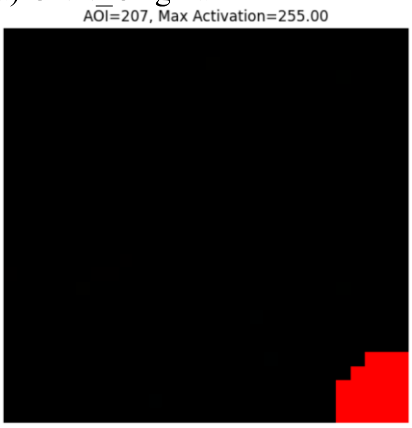
(a) CNN_Original



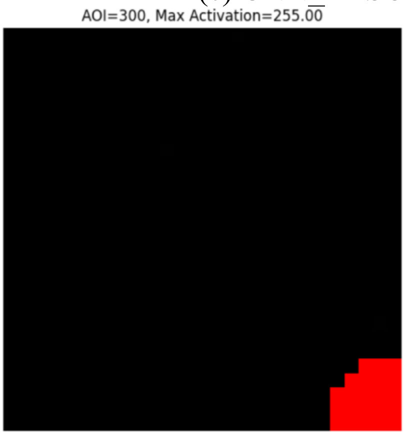
(b) CNN_PSO



(c) CNN_APSO



(d) CNN_WWO

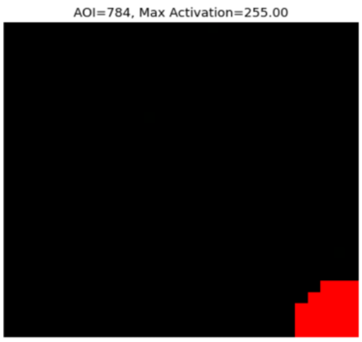


(e) CNN_APSWWO

(xviii) Grad-CAM of batch_normalization_4 layer of CNN

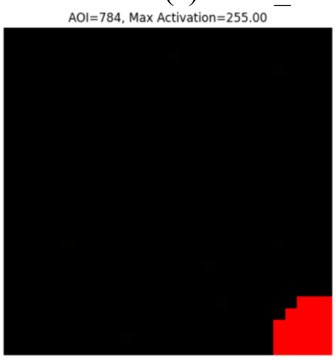
(a) CNN_Original

(b) CNN_PSO



(c) CNN_APSO

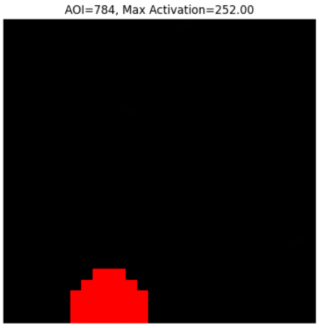
(d) CNN_WWO



(e) CNN_APSWWO

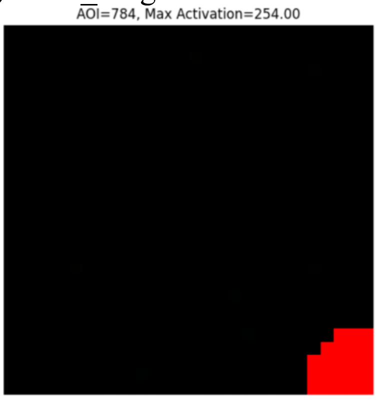
(xix) Grad-CAM of relu_3 layer of CNN

(a) CNN_Original

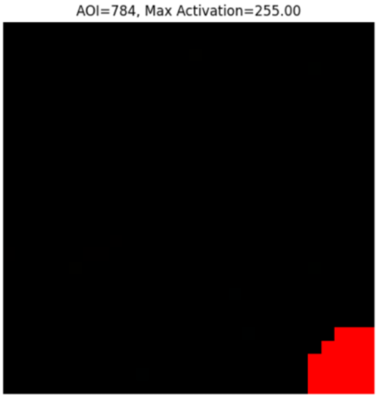


(b) CNN_PSO

(c) CNN_APSO

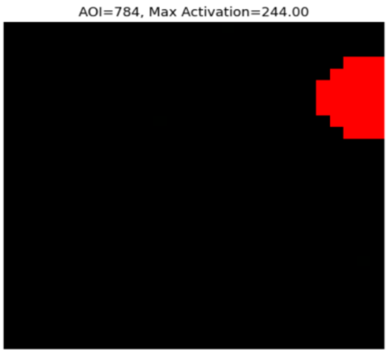


(d) CNN_WWO

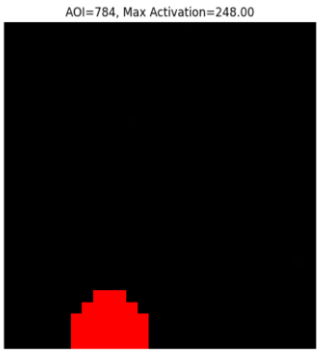


(e) CNN_APSWWO

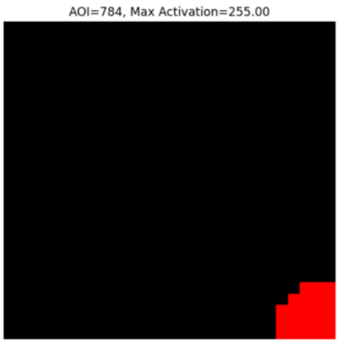
(xx) Grad-CAM of attention_1 layer of CNN



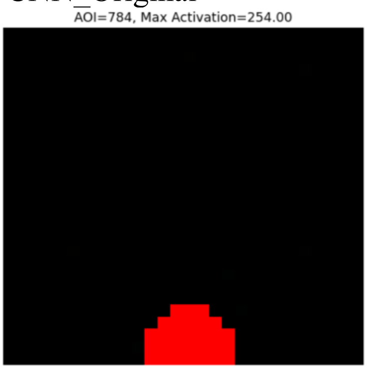
(a) CNN_Original



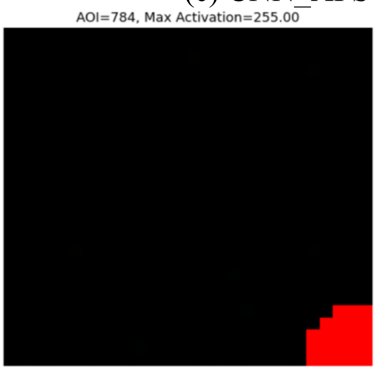
(b) CNN_PSO



(c) CNN_APSO

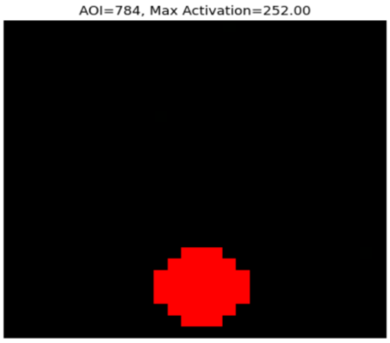


(d) CNN_WWO

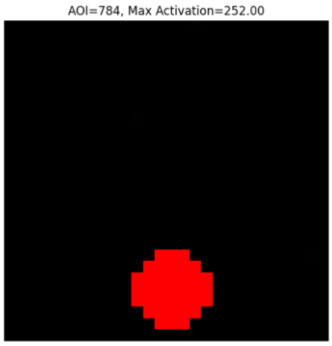


(e) CNN_APSWWO

(xxi) Grad-CAM of math.multifly_2 layer of CNN



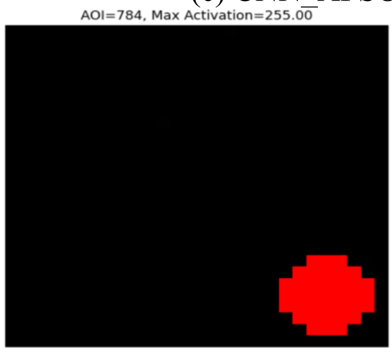
(a) CNN_Original



(b) CNN_PSO

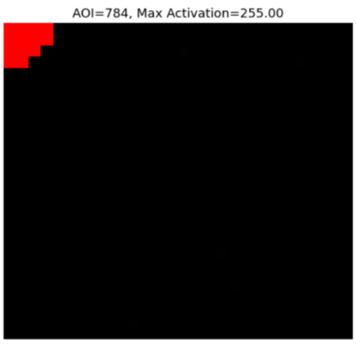
(c) CNN_APSO

(d) CNN_WWO

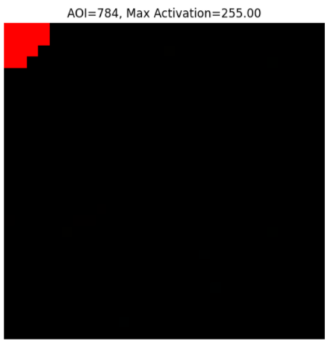


(e) CNN_APSWWO

(xxii) Grad-CAM of max_pooling2d_2 layer of CNN



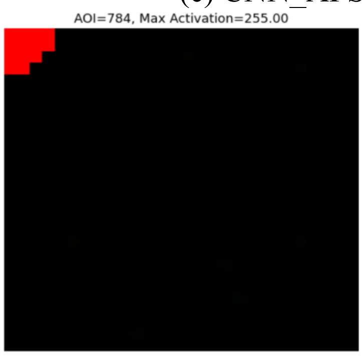
(a) CNN_Original



(b) CNN_PSO

(c) CNN_APSO

(d) CNN_WWO



(e) CNN_APSWWO

(xxiii) Grad-CAM of global_average_pooling2d layer of CNN

(a) CNN_Original

(b) CNN_PSO

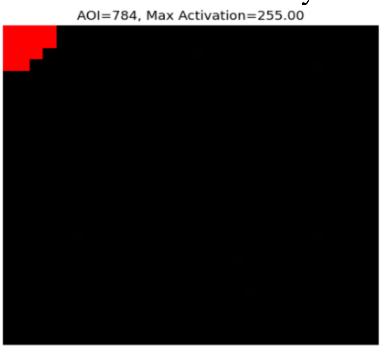
(c) CNN_APSO

(d) CNN_WWO

(e) CNN_APSWWO

(xxiv) Grad-CAM of dense layer of CNN

(a) CNN_Original



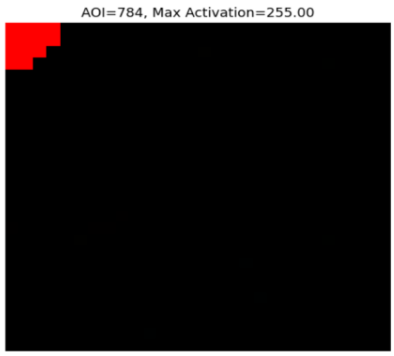
(b) CNN_PSO

(c) CNN_APSO

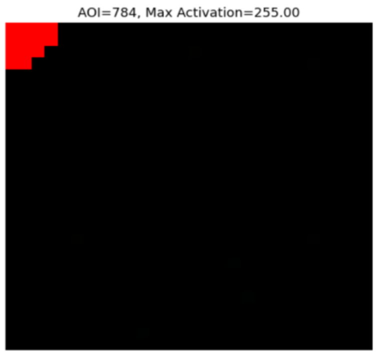
(d) CNN_WWO

(e) CNN_APSWWO

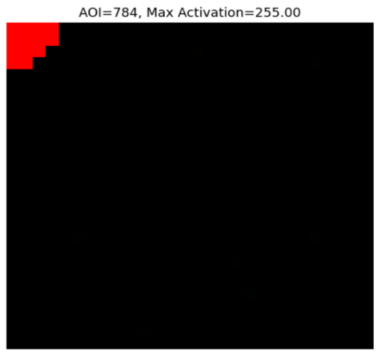
(xv) Grad-CAM of dropout layer of CNN



(a) CNN_Original

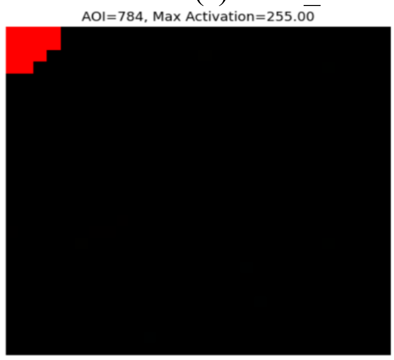


(b) CNN_PSO



(c) CNN_APSO

(d) CNN_WWO



(e) CNN_APSWWO

(xvi) Grad-CAM of dense_1 layer of CNN

Figure 4.66: Grad-CAM of 26-layer CNN Model

In Figure 4.66 and Table 4.23, the layer-wise Grad-CAM analysis of CNN is presented. Across various layers, each CNN model - CNN_Original, CNN_PSO, CNN_APSO, CNN_WWO, and CNN_APSWWO - demonstrates distinct performance in terms of the number of Areas of Interest (AOI) identified and their corresponding Maximum Activation values. In the 'conv2d' layer, CNN_APSO and CNN_APSWWO stand out with a significantly higher number of AOI compared to other models, indicating their ability to capture more nuanced features. In subsequent layers like 'batch_normalization' and 'conv2d_2', CNN_APSWWO consistently exhibits superior performance, showcasing a higher number of AOI. In the 'relu' layer, all CNN models display uniform performance with identical AOI counts and Maximum Activation values, suggesting a shared focus on specific image features at this stage of processing. Moving to deeper layers like 'conv2d_4' and 'batch_normalization_4', CNN_APSWWO consistently outperforms other models, highlighting its ability to discern intricate patterns within the input data.

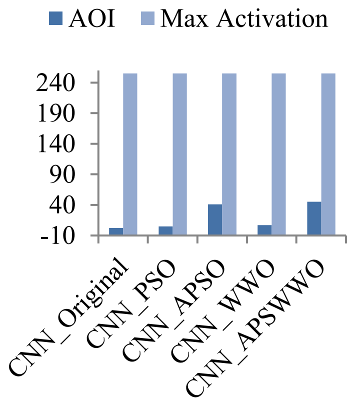
Table 4.23: Layer wise Grad-CAM value of CNN

Layer	**CNN_Original**		**CNN_PSO**		**CNN_APSO**		**CNN_WWO**		**CNN_APSWWO**	
	AOI	**Max Activ.**	**AOI**	**Max Activ.**	**AOI**	**Max Activ.**	**AOI**	**Max Activ.**	**AOI**	**Max Activ.**
conv2d	2	255	5	255	41	255	7	255	45	255
batch_normalization	210	255	365	255	491	255	389	255	600	255

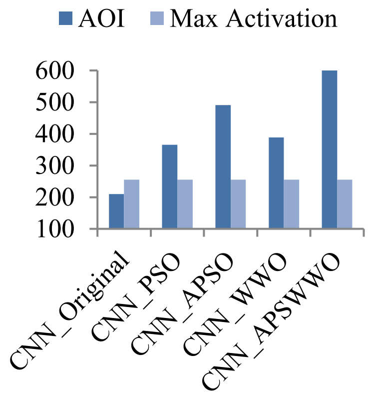
relu	784	255	784	255	784	255	784	255	784	255
con2d_1	37	255	39	255	130	255	54	255	207	255
batch_normalization_1	157	255	207	255	396	255	250	255	436	255
relu_1	784	255	784	255	784	255	784	255	752	255
conv2d_2	54	255	69	255	459	255	148	255	503	255
batch_normalization_2	43	255	127	255	199	255	148	255	356	255
relu_2	784	255	784	255	784	255	784	255	784	255
max_pooling2d	784	225	784	226	784	254	784	232	784	255
conv2d_3	428	202	607	211	702	239	626	221	703	239
batch_normalization_3	505	219	652	247	784	255	784	252	784	255
math.multifly	574	198	626	221	731	248	652	247	784	252
attention	304	226	574	198	658	247	629	204	696	248
math.multifly_1	784	218	784	229	784	240	784	233	784	243
max_pooling2d_1	784	239	784	242	784	251	784	250	784	254
conv2d_4	273	244	96	249	168	255	167	255	287	255
batch_normalization_4	36	255	168	255	274	255	207	255	300	255
relu_3	784	250	784	252	784	255	784	254	784	255
attention_1	784	250	784	252	784	255	784	254	784	255
math.multifly_2	784	244	784	248	784	255	784	254	784	255
max_pooling2d_2	784	252	784	255	784	255	784	254	784	255
global_average_pooling2d	784	255	784	255	784	255	784	255	784	255
dense	784	255	784	255	784	255	784	255	784	255
dropout	784	255	784	255	784	255	784	255	784	255
dense_1	784	255	784	255	784	255	784	255	784	255

In layers such as relu, relu_1, relu_2, global_average_pooling2d, dense, dropout, and dense_1, all techniques exhibit uniform performance with identical AOI (784) and Maximum Activation (255) values. This uniformity stems from the inherent characteristics of these layers within a neural network. Relu layers apply a consistent activation function with straightforward parameters, while the global_average_pooling2d layer performs a mathematical operation without learnable parameters, resulting in consistent outcomes. Dense layers maintain consistency once trained, as their learnable weights remain unchanged across models. Dropout layers, utilized for regularization, also yield uniform results due to consistent dropout rate settings. On the other hand, layers involving learnable parameters or complex operations, such as convolutional and batch normalization layers, exhibit more variation across models. Layers like max_pooling2d, math.multifly_1, max_pooling2d_1, relu_3, attention_1, math.multifly_2, and max_pooling2d_2 show identical AOI values across all techniques, as they primarily handle down-sampling and non-linear activation functions less influenced by variations in hyperparameters or network architectures. However, differences in maximum activation values can be attributed to CNN_APSWWO's superior performance in paddy disease classification, particularly its enhanced ability to activate and extract valuable features during training. This superior feature learning leads to higher maximum activation values, indicating CNN_APSWWO's proficiency in capturing and emphasizing important information in these layers.

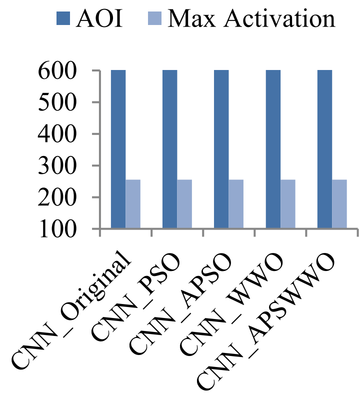
CNN_APSWWO consistently outperforms other techniques in layers such as conv2d, batch_normalization, conv2d_1, batch_normalization_1, conv2d_2, batch_normalization_2, conv2d_3, batch_normalization_3, math.multifly, attention, math.multifly_1, conv2d_4, and batch_normalization_4. Its higher accuracy across all layers suggests its effective learning and utilization of crucial features and patterns in the data, contributing to its superior performance. The consistently higher accuracy of CNN_APSWWO likely plays a pivotal role in its outstanding performance across multiple layers compared to other techniques. The graphical representation of Grad-CAM values for each CNN technique is depicted in Figure 4.67.



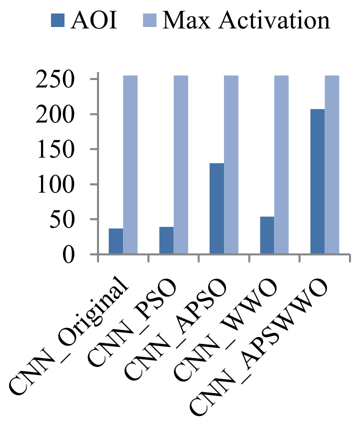
(a) conv2d



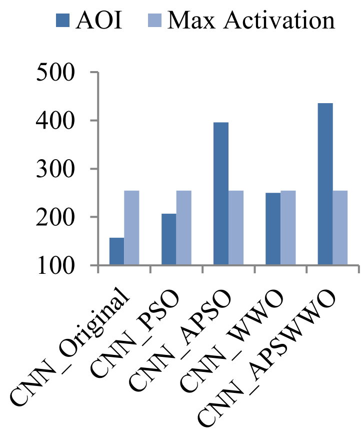
(b) batch_normalization



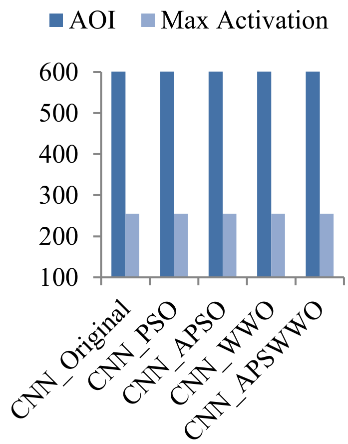
(c) relu



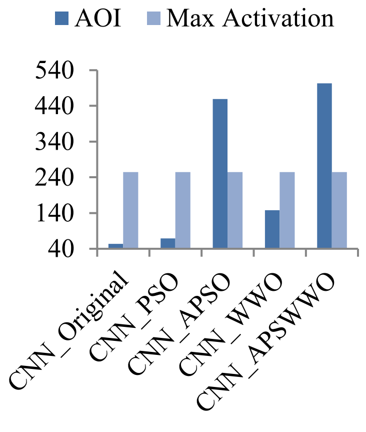
(d) con2d_1



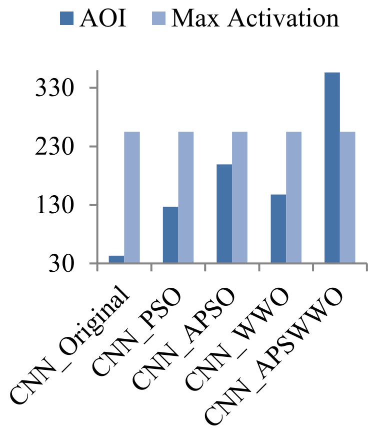
(e) batch_normalization_1



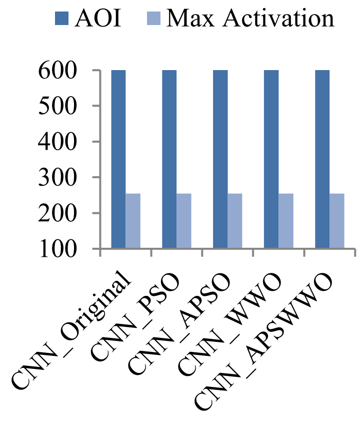
(f) relu_1



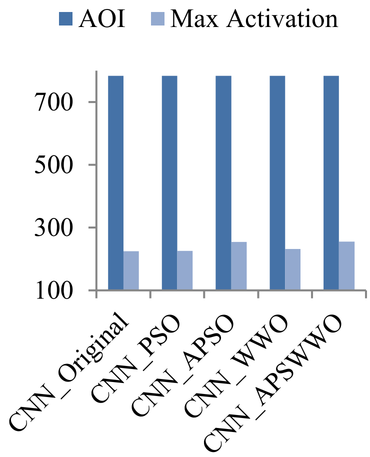
(g) conv2d_2



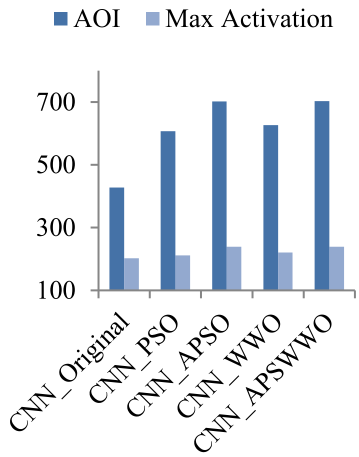
(h) batch_normalization_2



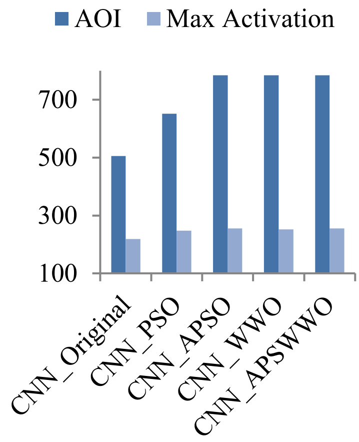
(i) relu_2



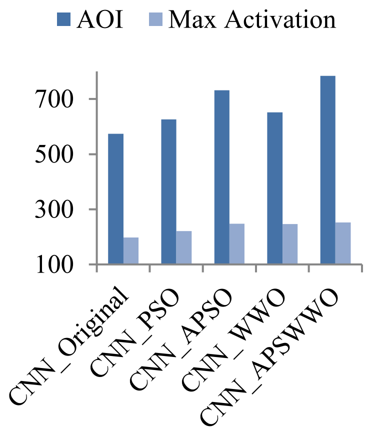
(j) max_pooling2d



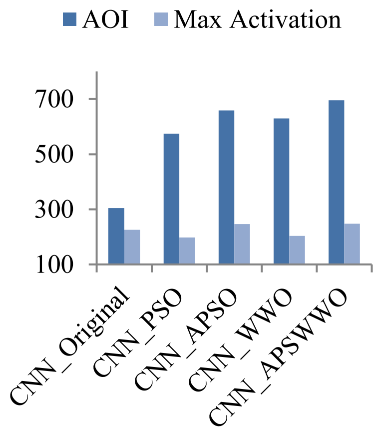
(k) conv2d_3



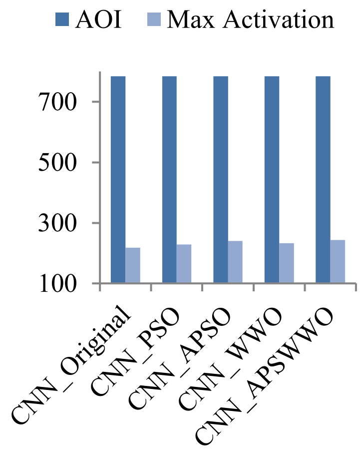
(l) batch_normalization_3



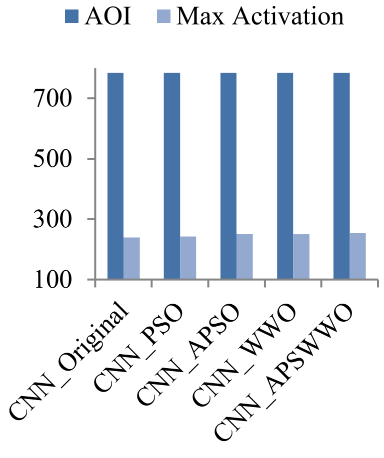
(m) math.multifly



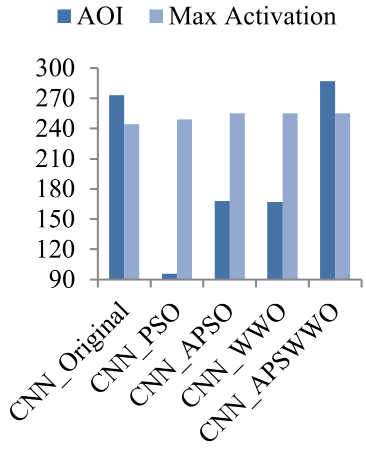
(n) attention



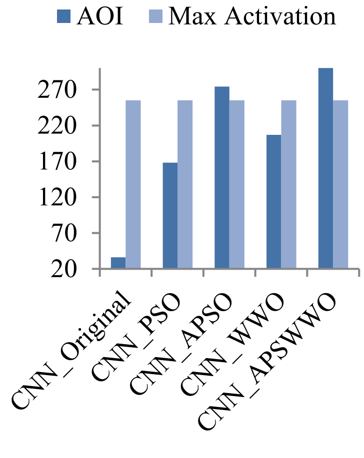
(o) math.multifly_1



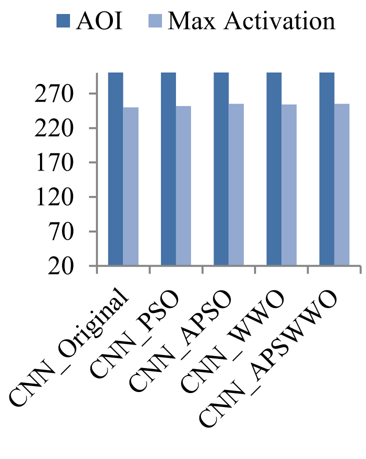
(p) max_pooling2d_1



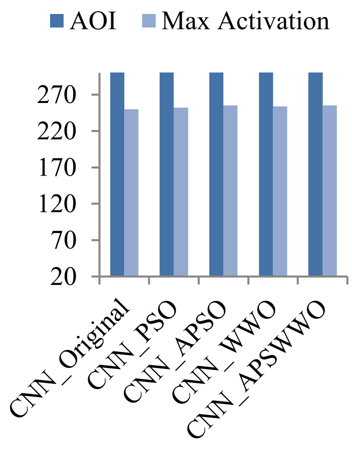
(q) conv2d_4



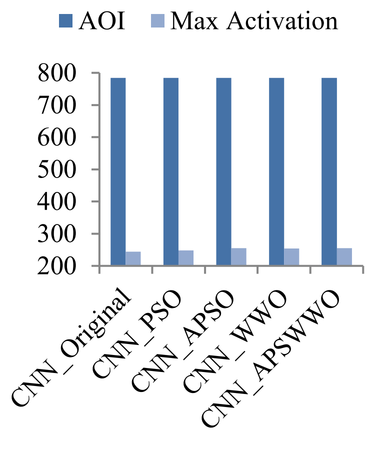
(r) batch_normalization_4



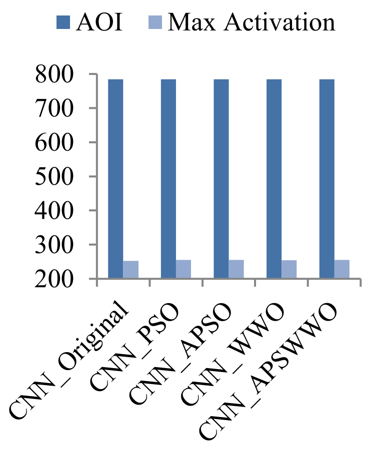
(s) relu_3



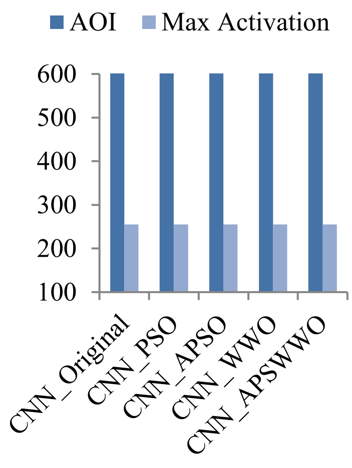
(t) attention_1



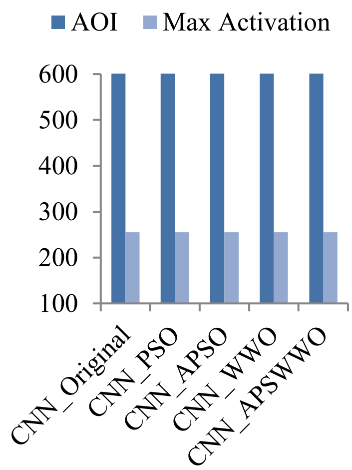
(u) math.multifly_2



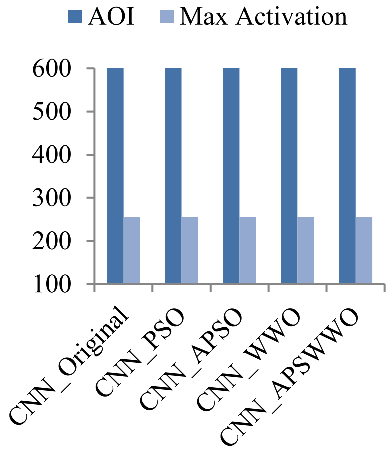
(v) max_pooling2d_2



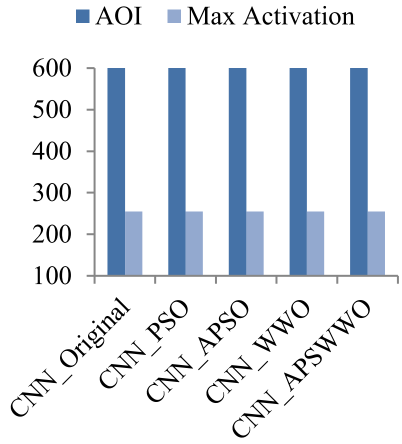
(w) global_average_pooling2d



(x) dense



(y) dropout



(z) dense_1

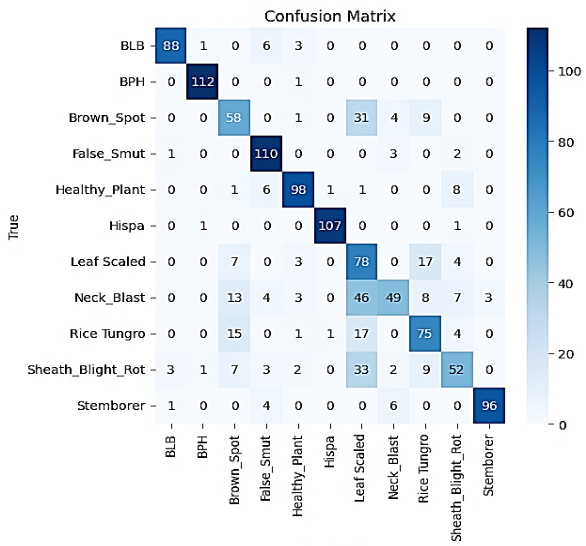
Figure 4.67: Visualization of Grad-CAM values of CNN

4.4.2 Light Weight Vision Transformer

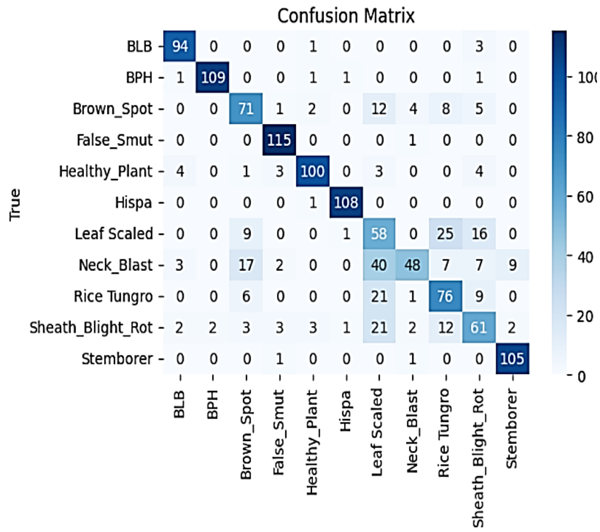
This section conducts a comparative analysis among five different LiteViT models: LiteViT_Original, LiteViT_PSO, LiteViT_APSO, LiteViT_WWO, and LiteViT_APSWWO, in their classification of paddy diseases into categories including BLB, BPH, brown spot, false smut, healthy plant, hispa, leaf scaled, neck blast, tungro, sheath blight rot, and stemborer. The aim of this analysis is to assess and contrast the performance of these different techniques. Each of these LiteViT techniques employs distinct hyperparameter configurations, as summarized in Table 4.24. LiteViT_Original uses a learning rate of 0.01, a batch size of 32, 100 epochs, 4 attention heads, 64 projection dimensions, and dropout rates of 0.4 for the LiteViT layers, 0.5 for the Attention layer, and 0.2 for Attention_1. On the other hand, LiteViT_PSO utilizes a learning rate of 0.001, a batch size of 166, 122 epochs, eight attention heads, a projection dimension of 100, and dropout rates of 0.4, 0.2, and 0.3 for the LiteViT, Attention, and Attention_1 layers, respectively. LiteViT_APSO employs a learning rate of 0.001, a batch size of 134, 72 epochs, 64 attention heads, a projection dimension of 252, and dropout rates of 0.4, 0.2, and 0.6 for the corresponding layers. LiteViT_WWO uses a learning rate of 0.004, a batch size of 76, 67 epochs, 16 attention heads, a projection dimension of 95, and dropout rates of 0.5, 0.5, and 0.2 for the LiteViT, Attention, and Attention_1 layers. Lastly, LiteViT_APSWWO is configured with a learning rate of 0.001, a batch size of 256, 287 epochs, 77 attention heads, a projection dimension of 512, and dropout rates of 0.5, 0.2, and 0.4 for the respective layers. These hyperparameters play a critical role in configuring and training each LiteViT technique, influencing their performance in the classification of paddy diseases. Figure 4.68, Figure 4.69, Figure 4.70, and Figure 4.72 represents the confusion matrix, ROC curve, precision-recall curve, and learning curves of LiteViT_Original, LiteViT_PSO, LiteViT_APSO, LiteViT_WWO, and LiteViT_APSWWO.

Table 4.24: Optimized values of LiteViT

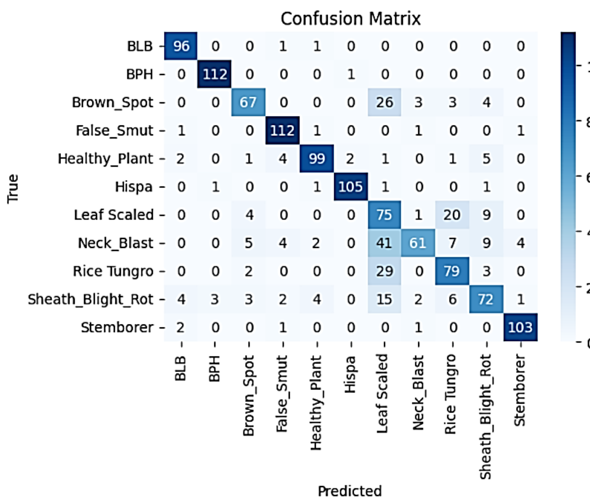
LiteViT Techniques	Learning rate	Batch size	Epoch number	Attention heads	Projection dimension	Dropout rate for CNN	Dropout rate for Attention layer	Dropout rate for Attention_1 layer
LiteViT_Original	0.01	32	100	4	64	0.4	0.5	0.2
LiteViT_PSO	0.001	166	122	8	100	0.4	0.2	0.3
LiteViT_APSO	0.001	134	72	64	252	0.4	0.2	0.6
LiteViT_WWO	0.004	76	67	16	95	0.5	0.5	0.2
LiteViT_APSWWO	0.001	256	287	77	512	0.5	0.2	0.4



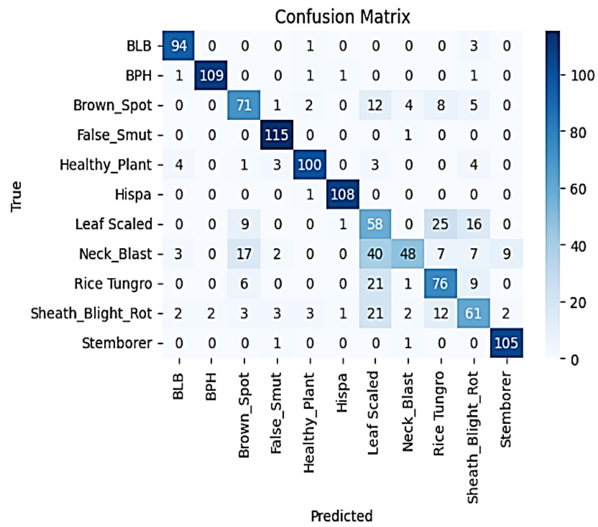
(a) LiteViT_Original



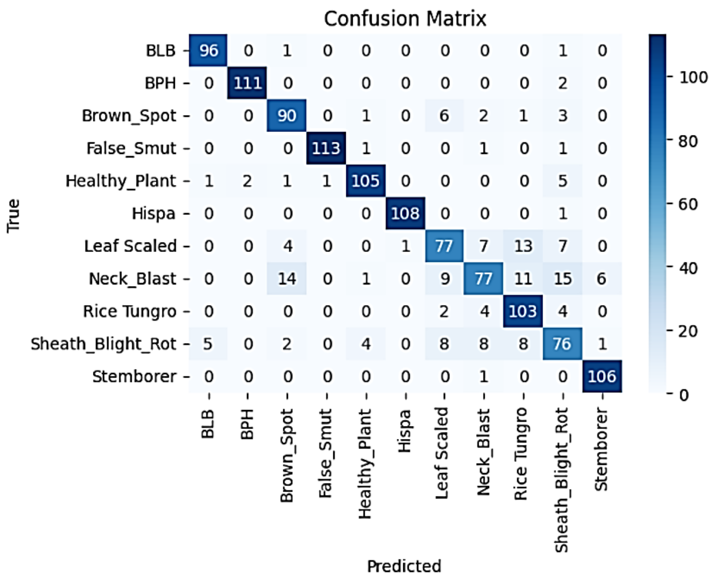
(b) LiteViT_PSO



(c) LiteViT_APSO



(d) LiteViT_WWO



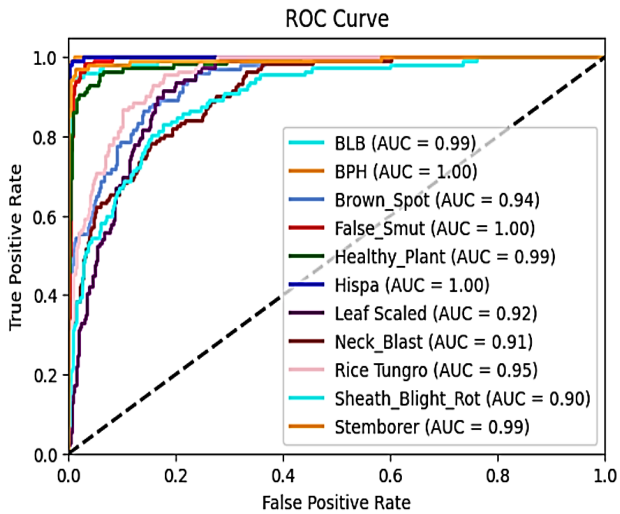
(e) LiteViT_APSWWO

Figure 4.68: Confusion matrix of LiteViT

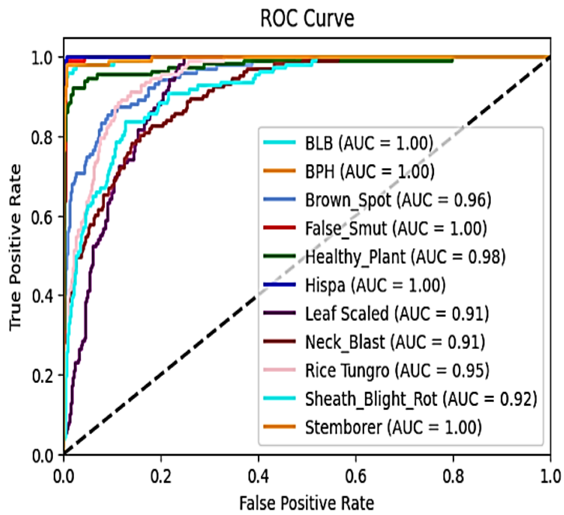
Table 4.25 presents a comprehensive comparative analysis of five LiteViT models: LiteViT_Original, LiteViT_PSO, LiteViT_APSO, LiteViT_WWO, and LiteViT_APSWWO, focusing on their performance in classifying paddy diseases. These models are assessed based on a range of critical performance metrics, shedding light on their strengths and weaknesses. In terms of correctly classified instances, LiteViT_APSWWO emerges as the top performer, with an impressive average of 96.55 instances correctly classified. This metric signifies the model's ability to effectively identify the different disease categories within the dataset. In contrast, LiteViT_Original achieves the lowest average in this category, classifying 83.91 instances correctly on average. Misclassification is another crucial aspect of model evaluation. LiteViT_APSWWO once again shines by demonstrating the lowest average misclassified instances at 15.09, indicating its superior accuracy in disease classification. On the contrary, LiteViT_Original records the highest average misclassified instances at 27.73.

When considering metrics like recall, precision, and F1-score, which measure the model's ability to correctly identify positive cases and minimize false positives, LiteViT_APSWWO consistently ranks highest. It achieves the highest recall, precision, and F1-score, showcasing its effectiveness in disease classification. In contrast, LiteViT_Original falls slightly behind in these aspects, underlining its relative shortcomings. Additionally, LiteViT_APSWWO maintains the lowest training loss and testing loss, indicating its superior performance in learning from the training data and generalizing to unseen data. ROC-AUC scores, which signify the model's discrimination ability, also highlight the excellence of LiteViT_APSWWO. It achieves the highest ROC-AUC score among all models. Moreover, LiteViT_APSWWO stands out with the highest Average Precision (AP) value, underscoring its effectiveness in classifying positive instances while minimizing false positives.

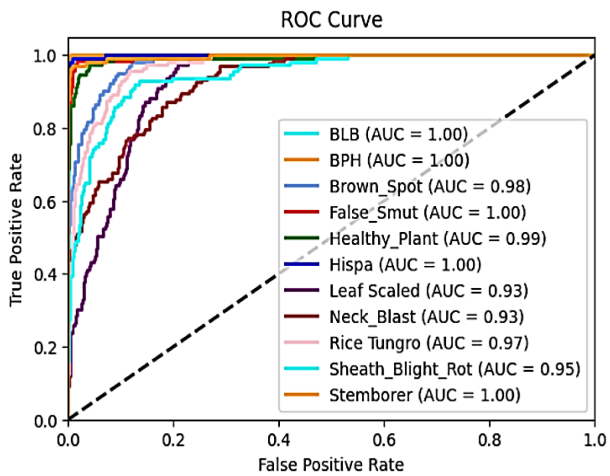
Error rates and accuracy percentages provide a comprehensive view of overall model performance. LiteViT_APSWWO once again excels with the lowest error rate and the highest accuracy at 14% and 86%, respectively. This indicates that LiteViT_APSWWO consistently outperforms the other models in correctly classifying paddy diseases. Figure 104 highlights that LiteViT_APSWWO is the top performer across various critical performance metrics, making it the model of choice for paddy disease classification. It consistently demonstrates superior performance in correctly classifying instances, minimizing misclassifications, achieving high recall, precision, and F1-score, and effectively learning from training data and generalizing to unseen data. LiteViT_Original, on the other hand, ranks lower in terms of performance metrics, emphasizing the advancements achieved by LiteViT models with optimization techniques like PSO, APSO, WWO, and APSWWO.



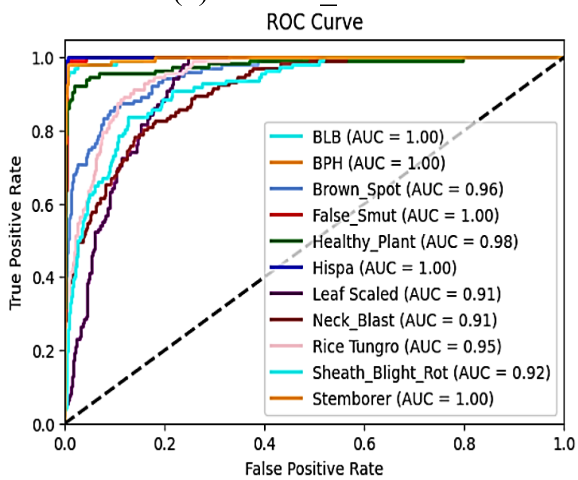
(a) LiteViT_Original



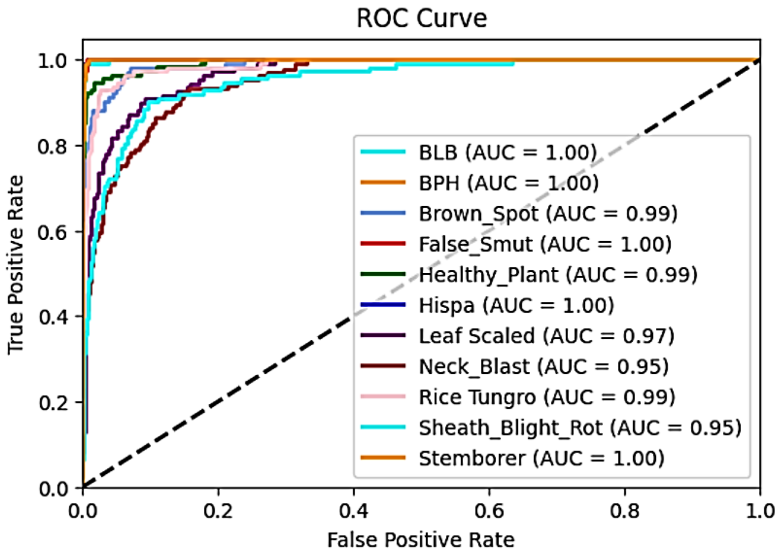
(b) LiteViT_PSO



(c) LiteViT_APSO

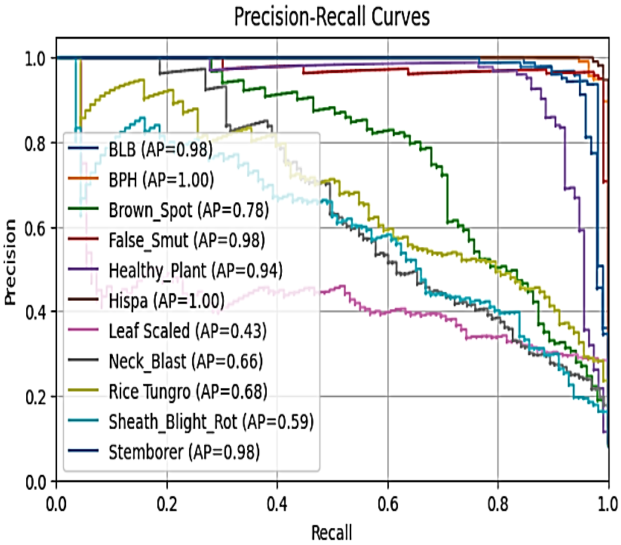


(d) LiteViT_WWO

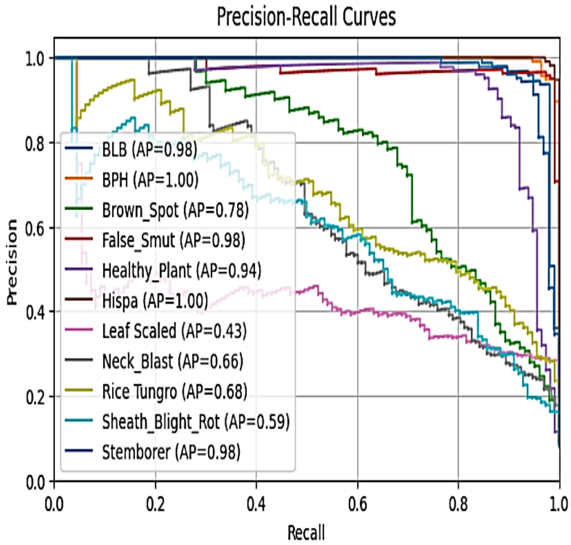


(e) LiteViT_APSWWO

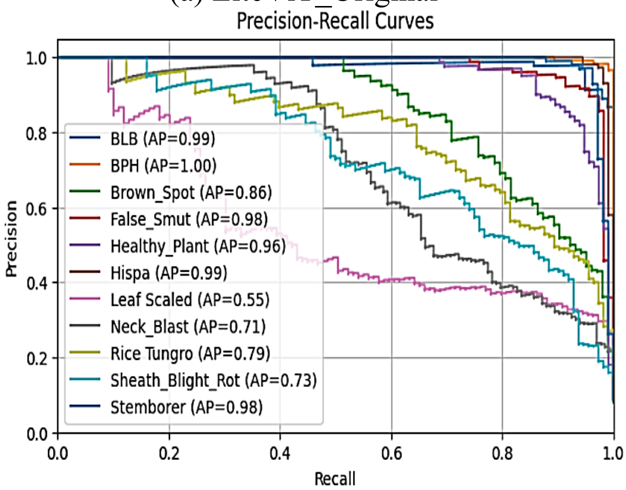
Figure 4.69:ROC curve of LiteViT



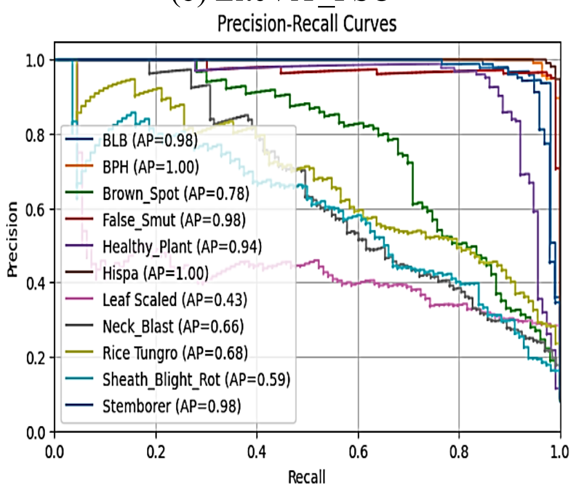
(a) LiteViT_Original



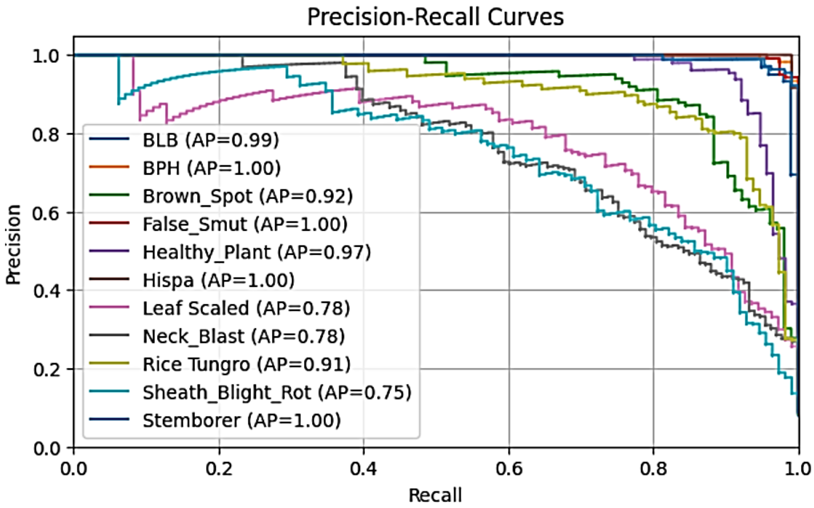
(b) LiteViT_PSO



(c) LiteViT_APSO



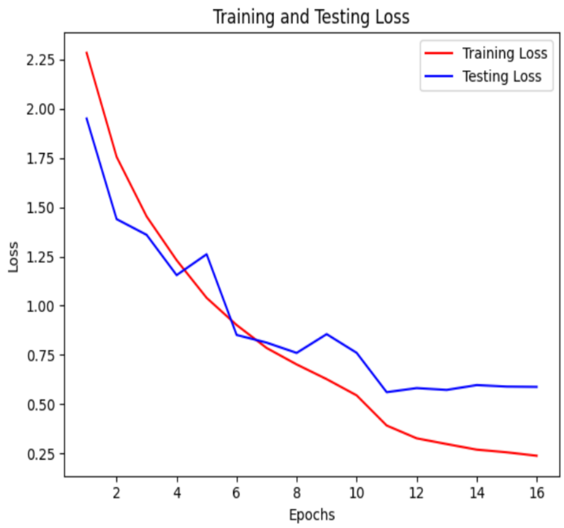
(d) LiteViT_WWO



(e) LiteViT_APSWWO

Figure 4.70: Precision-recall curve of LiteViT

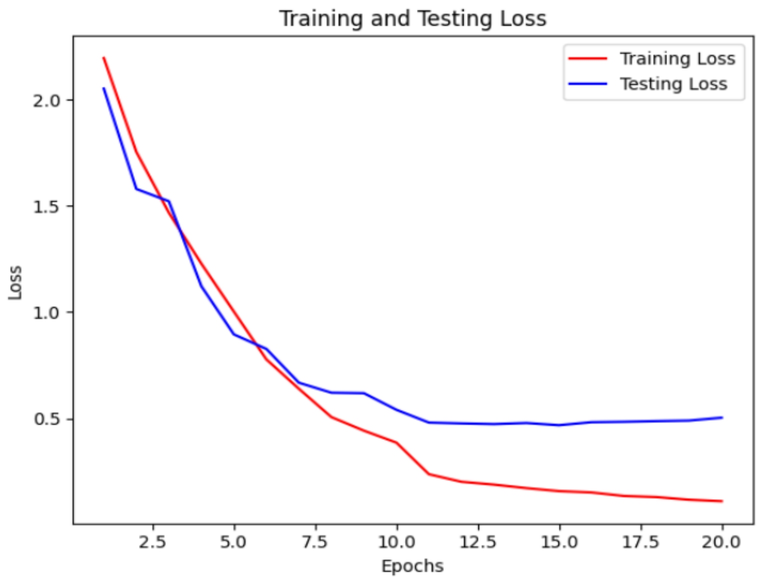
(a) LiteViT_Original



(b) LiteViT_PSO

(c) LiteViT_APSO

(d) LiteViT_WWO

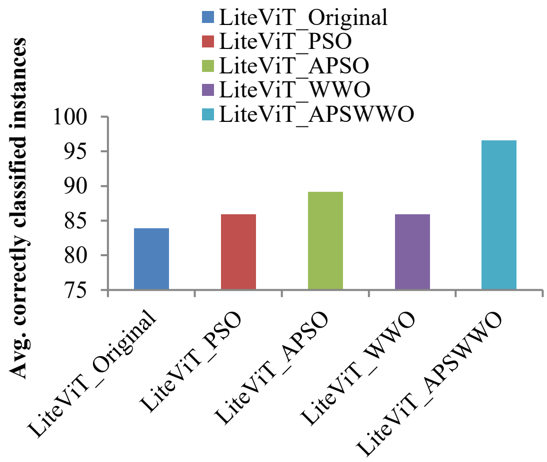


(e) LiteViT_APSWWO

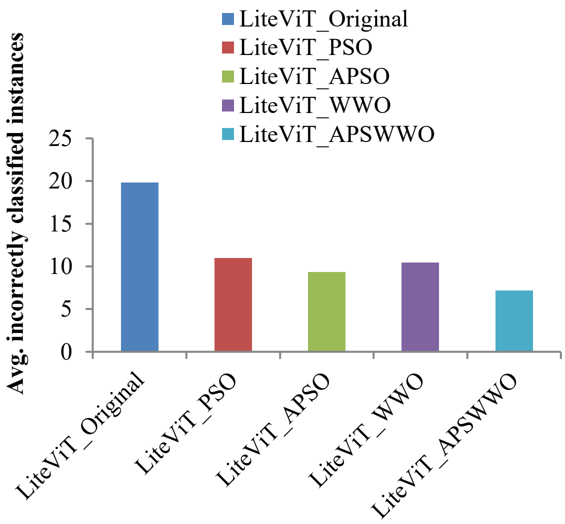
Figure 4.71: Learning curve of LiteViT

Table 4.25: Performance metrics of LiteViT

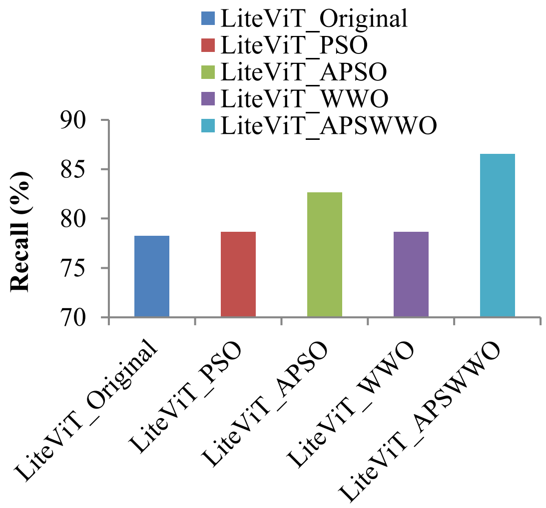
Performance Metrics	LiteViT_Original	LiteViT_PSO	LiteViT_APSO	LiteViT_WWO	LiteViT_APSWWO
Avg. correctly classified instances	83.91	85.91	89.18	85.91	96.55
Avg. incorrectly classified instances	27.73	25.55	22.45	25.73	15.09
Recall (%)	78.27	78.64	82.64	78.64	86.55
Precision (%)	75.82	77.64	80.55	77.64	87
F1-score (%)	75.64	77.18	80.55	77.18	86.64
Training loss	0.3744	0.3274	0.2387	0.3274	0.1097
Testing loss	0.7435	0.7532	0.5883	0.7532	0.5026
ROC-AUC score	0.963	0.966	0.893	0.966	0.985
AP value	0.814	0.82	0.867	0.82	0.918
Error rate (%)	25	22	20	23	14
Accuracy (%)	75	78	80	77	86



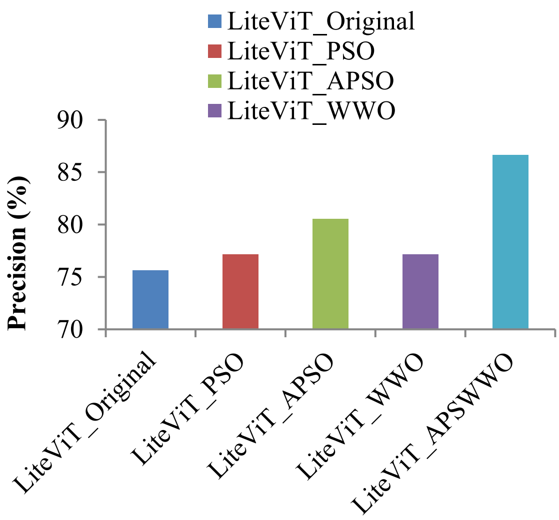
(a) Avg. correctly classified instances



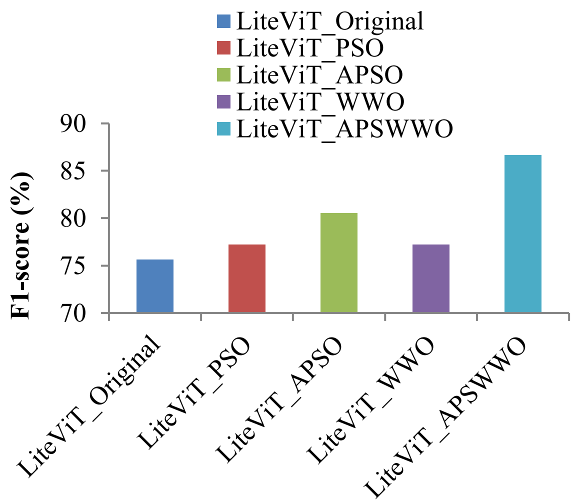
(b) Avg. incorrectly classified instances



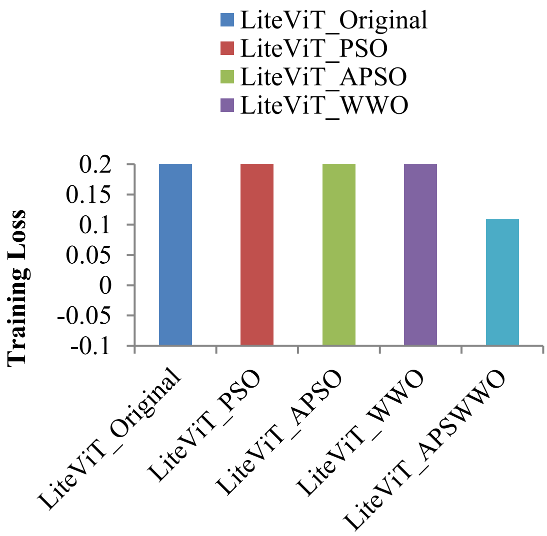
(c) Recall



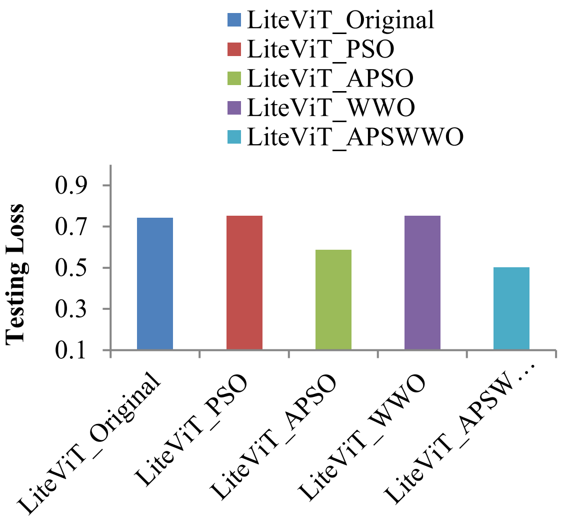
(d) Precision



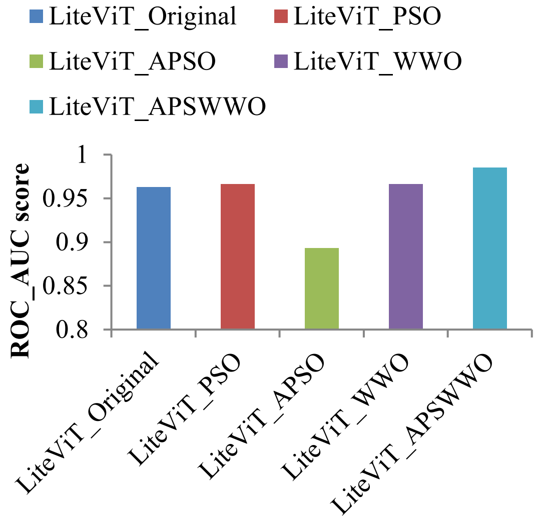
(e) F1-score



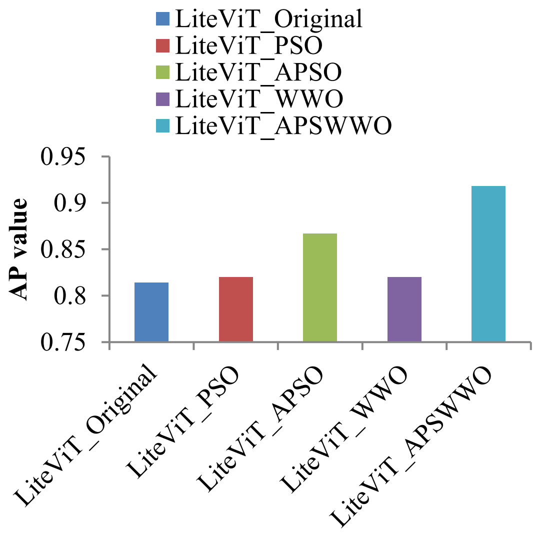
(f) Training Loss



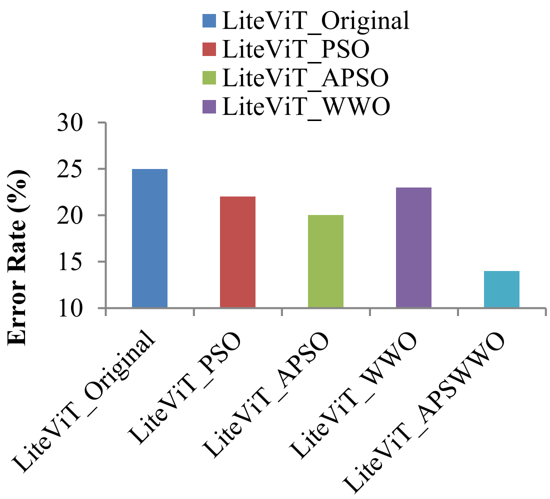
(g) Testing Loss



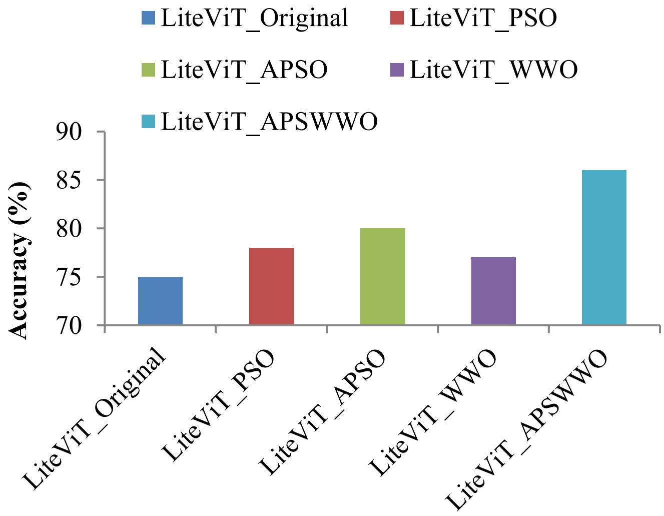
(h) ROC-AUC score



(i) Average Precision value

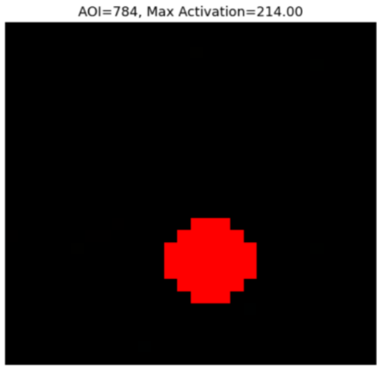


(j) Error Rate

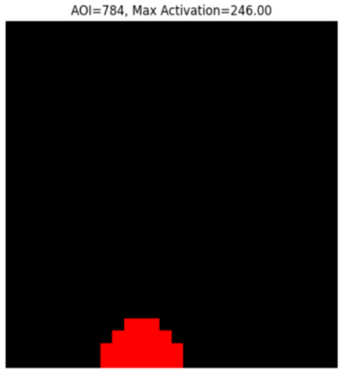


(k) Accuracy

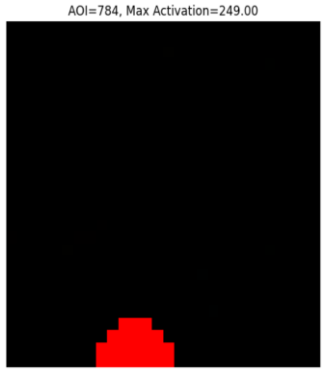
Figure 4.72: Performance metrics of LiteViT



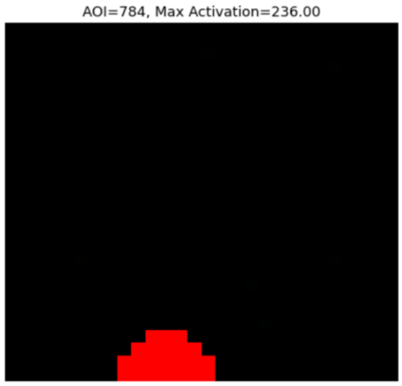
(a) LiteViT_Original



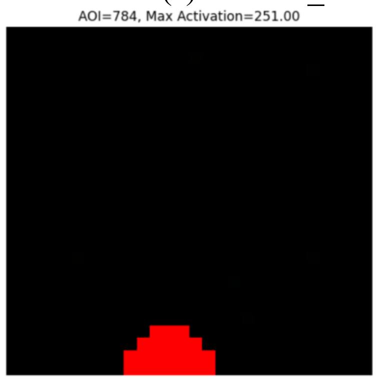
(b) LiteViT_PSO



(c) LiteViT_APSO

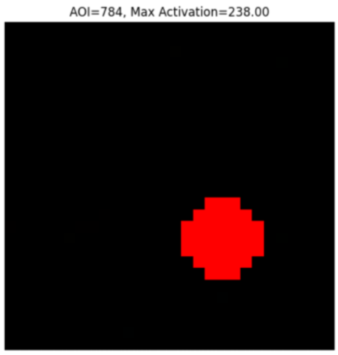


(d) LiteViT_WWO

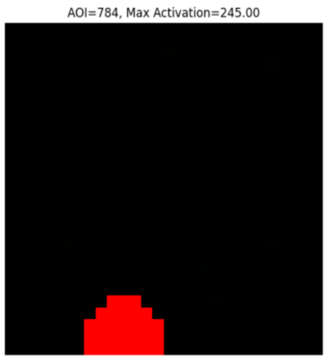


(e) LiteViT_APSWWO

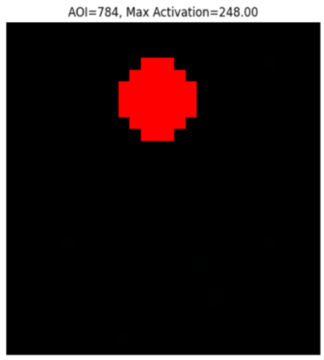
(i) Grad-CAM of conv2d layer of LiteViT



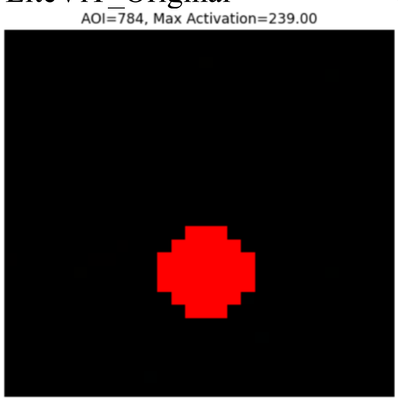
(a) LiteViT_Original



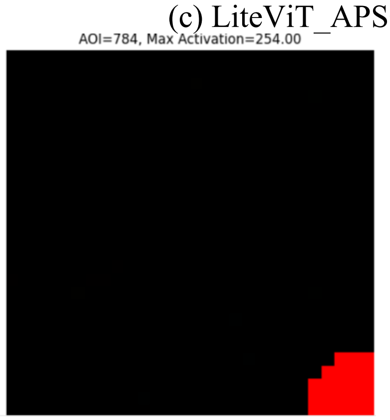
(b) LiteViT_PSO



(c) LiteViT_APSO

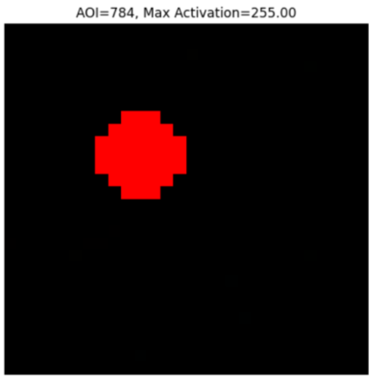


(d) LiteViT_WWO

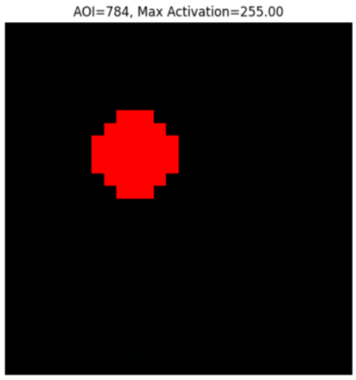


(e) LiteViT_APSWWO

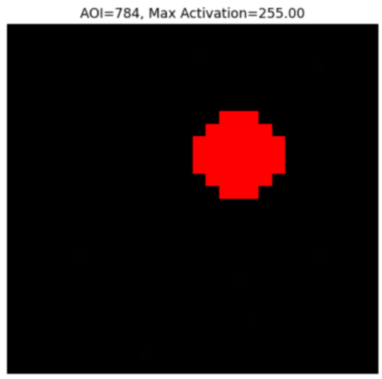
(ii) Grad-CAM of max_pooling2d layer of LiteViT



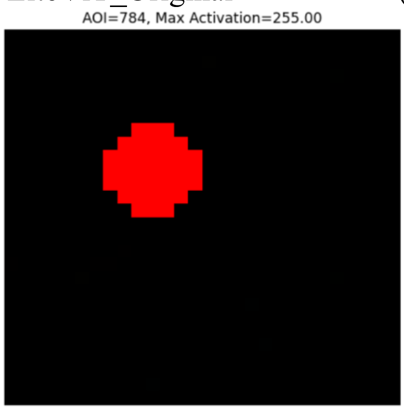
(a) LiteViT_Original



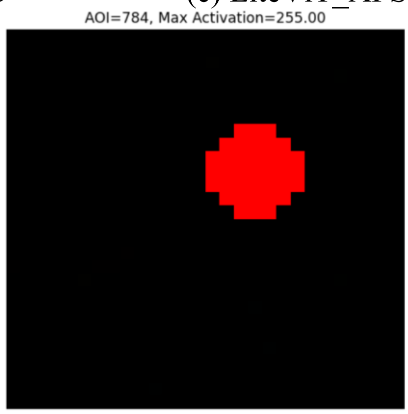
(b) LiteViT_PSO



(c) LiteViT_APSO



(d) LiteViT_WWO



(e) LiteViT_APSWWO

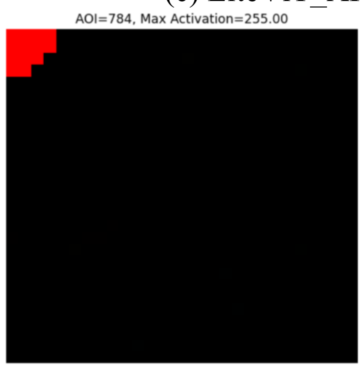
(iii) Grad-CAM of conv2d_1 layer of LiteViT

(a) LiteViT_Original

(b) LiteViT_PSO

(c) LiteViT_APSO

(d) LiteViT_WWO



(e) LiteViT_APSWWO

(iv) Grad-CAM of max_pooling2d_1 layer of LiteViT

(a) LiteViT_Original

(b) LiteViT_PSO

(c) LiteViT_APSO

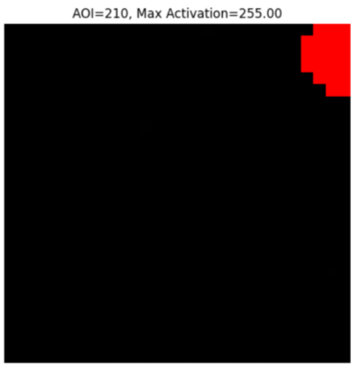
(d) LiteViT_WWO

(e) LiteViT_APSWWO

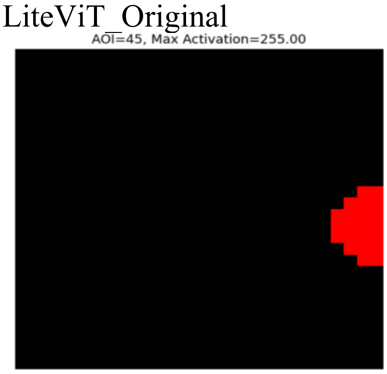
(v) Grad-CAM of conv2d_2 layer of LiteViT

(a) LiteViT_Original

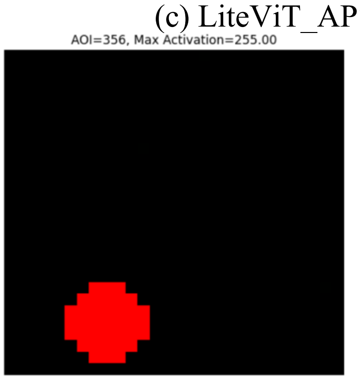
(b) LiteViT_PSO



(c) LiteViT_APSO

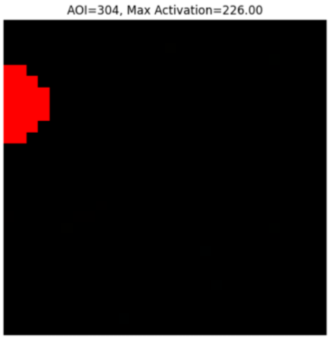


(d) LiteViT_WWO

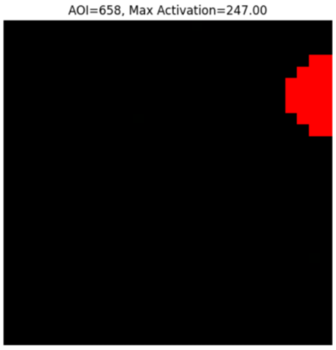


(e) LiteViT_APSWWO

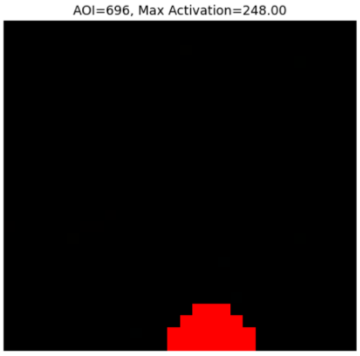
(vi) Grad-CAM of attention layer of LiteViT



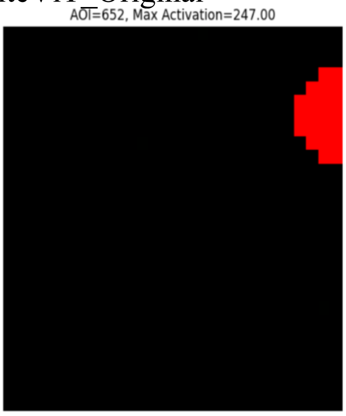
(a) LiteViT_Original



(b) LiteViT_PSO



(c) LiteViT_APSO

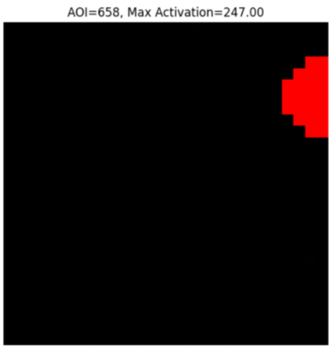


(d) LiteViT_WWO

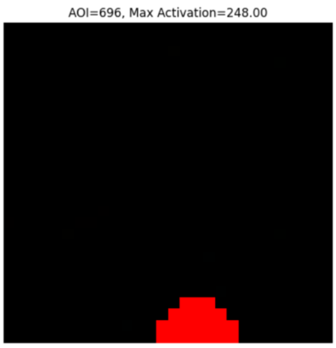
(e) LiteViT_APSWWO

(vii) Grad-CAM of operators.add layer of LiteViT

(a) LiteViT_Original

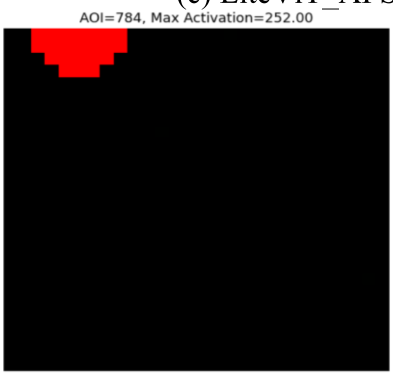


(b) LiteViT_PSO



(c) LiteViT_APSO

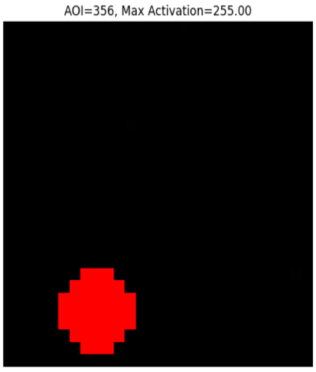
(d) LiteViT_WWO



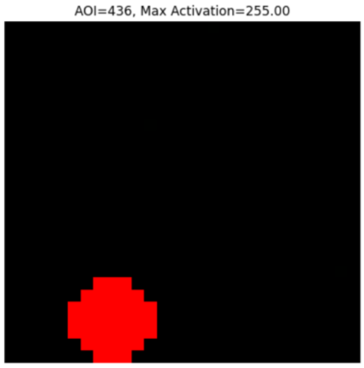
(e) LiteViT_APSWWO

(viii) Grad-CAM of attention_1 layer of LiteViT

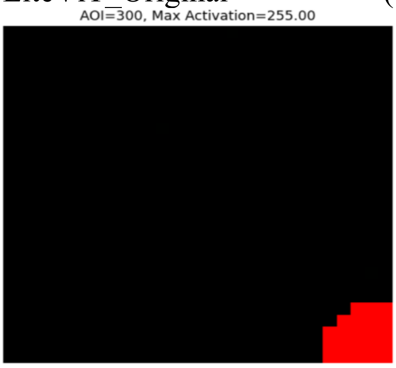
(a) LiteViT_Original



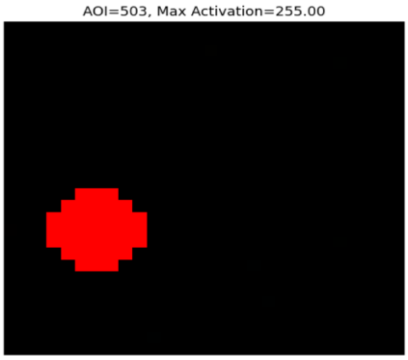
(b) LiteViT_PSO



(c) LiteViT_APSO



(d) LiteViT_WWO



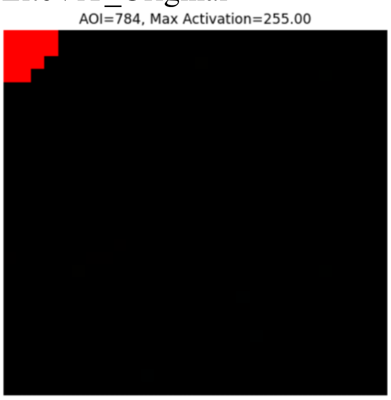
(e) LiteViT_APSWWO

(ix) Grad-CAM of operators.add_1 layer of LiteViT

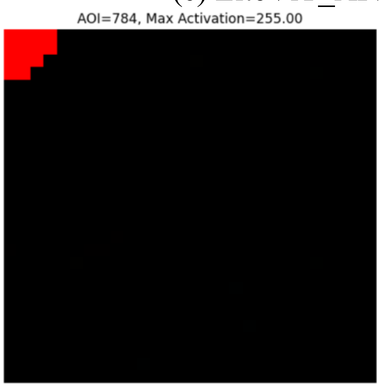
(a) LiteViT_Original

(b) LiteViT_PSO

(c) LiteViT_APSO



(d) LiteViT_WWO



(e) LiteViT_APSWWO

(x) Grad-CAM of global_average_pooling2d layer of LiteViT

(a) LiteViT_Original

(b) LiteViT_PSO

(c) LiteViT_APSO

(d) LiteViT_WWO

(e) LiteViT_APSWWO

(xi) Grad-CAM of dense layer of LiteViT

(a) LiteViT_Original

(b) LiteViT_PSO

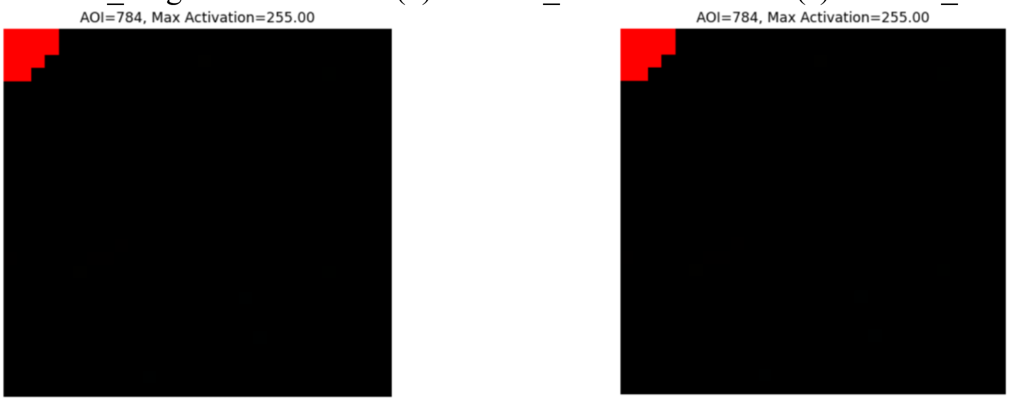
(c) LiteViT_APSO

(d) LiteViT_WWO

(e) LiteViT_APSWWO

(xii) Grad-CAM of dropout layer of LiteViT

(a) LiteViT_Original (b) LiteViT_PSO (c) LiteViT_APSO



(d) LiteViT_WWO (e) LiteViT_APSWWO

(xiii) Grad-CAM of dense_1 layer of LiteViT

Figure 4.73: Grad-CAM of LiteViT model

The Table 4.26 and Figure 4.73 provide a comprehensive overview of the Grad-CAM analysis for each layer across different LiteViT models, including LiteViT_Original, LiteViT_PSO, LiteViT_APSO, LiteViT_WWO, and LiteViT_APSWWO. Grad-CAM visualizations help us understand the Areas of Interest (AOI) and Maximum Activation values for various layers in these models, shedding light on their decision-making processes in image classification. In the 'conv2d' and 'max_pooling2d' layers, the AOI sizes are the same for all models, indicating that these layers maintain a broad focus covering the entire input image. However, the differences in Maximum Activation values suggest that although these layers focus on the entire image, they may assign varying degrees of importance to different regions. For example, LiteViT_APSWWO has the highest Maximum Activation value, indicating a more pronounced emphasis on specific regions despite the broad AOI. The 'attention' layer is intriguing because it exhibits variations in AOI size, indicating that different LiteViT models emphasize different spatial regions during this layer. Despite the distinct AOIs, the Maximum Activation values remain the same. This suggests that, while the models focus on different areas, the intensity of attention within these regions is consistent, signifying their significance in the decision-making process. 'conv2d_1,' 'max_pooling2d_1,' and 'conv2d_2' layers show consistent AOI sizes and Maximum Activation values across all algorithms. This uniformity may be attributed to the layers' inherent characteristics and their role in processing input images.

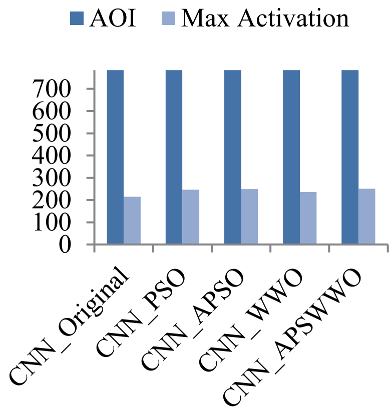
As for the last four layers, 'global_average_pooling2d,' 'dense,' 'dropout,' and 'dense_1,' the Grad-CAM values are identical across all algorithms. These layers may not significantly affect the model's spatial attention or feature activations, leading to consistent results. This consistency implies that the models focus on the entire input image during these layers, with a strong emphasis on the input's global characteristics rather than specific regions. The variations in AOI and Maximum Activation values across layers and algorithms highlight the intricate decision-making processes within LiteViT models. These patterns provide insights into how different models focus on specific regions and features during image classification tasks, ultimately contributing to their varied performance in these tasks. The consistent values in the last four layers suggest a commonality in their role and influence on the model's decision-making, potentially emphasizing global image characteristics. The graphical representation of Grad-CAM value for each LiteViT techniques has been depicted in Figure 4.74.

In a comparative analysis among LiteViT_Original, LiteViT_PSO, LiteViT_APSO, LiteViT_WWO, and LiteViT_APSWWO, we can observe that each model shares similar AOI characteristics in most layers, emphasizing the full image area. However, there are variations in Maximum Activation values, indicating differences in the importance assigned to specific regions within these layers. LiteViT_APSWWO consistently shows the highest Maximum Activation values, suggesting that it places the most importance on specific regions during multiple layers. In terms of overall performance, LiteViT_APSWWO stands out, as it consistently exhibits the highest Maximum Activation values across various layers. This suggests that LiteViT_APSWWO pays close attention to specific regions throughout its decision-making process, potentially contributing to its superior performance in image classification tasks.

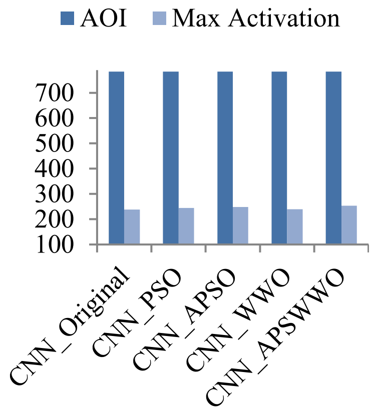
Table 4.26: Layer wise Grad-CAM value of LiteViT

Layer	LiteViT_Original		LiteViT_PSO		LiteViT_APSO		LiteViT_WWO		LiteViT_APSWWO	
	AOI	Max Activ.	AOI	Max Activ.	AOI	Max Activ.	AOI	Max Activ.	AOI	Max Activ.
conv2d	784	214	784	246	784	249	784	236	784	251
max_pooling2d	784	238	784	245	784	248	784	239	784	254
conv2d_1	784	255	784	255	784	255	784	255	784	255
max_pooling2d_1	784	255	784	255	784	255	784	255	784	255
conv2d_2	784	255	784	255	784	255	784	255	784	255
attention	37	255	157	255	210	255	45	255	356	255
operators.add	304	226	658	247	696	248	652	247	784	252
attention_1	629	204	658	247	696	248	626	221	784	252
operators.add_1	207	255	356	255	436	255	300	255	503	255
global_average_pooling2d	784	255	784	255	784	255	784	255	784	255
dense	784	255	784	255	784	255	784	255	784	255

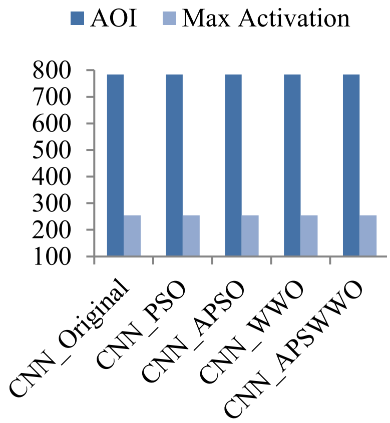
dropout	784	255	784	255	784	255	784	255	784	255
dense_1	784	255	784	255	784	255	784	255	784	255



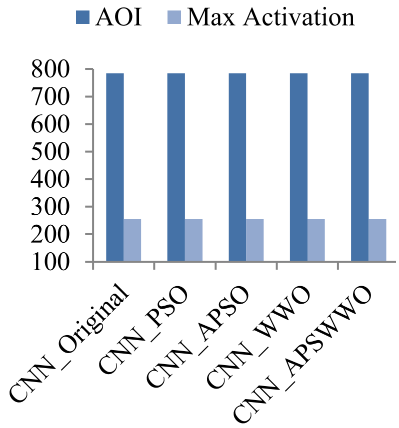
(a) conv2d



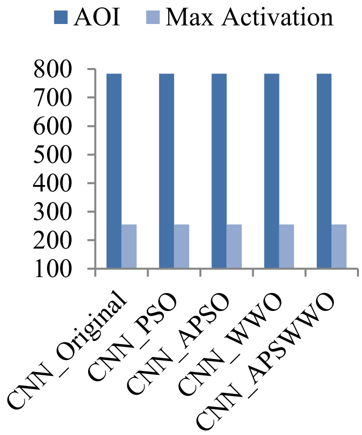
(b) max_pooling2d



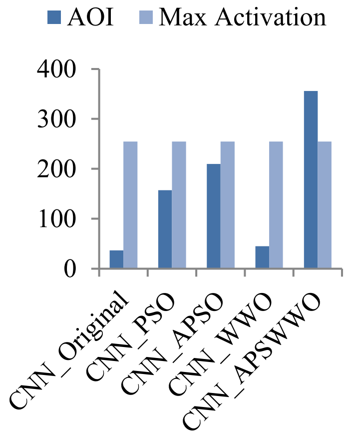
(c) conv2d_1



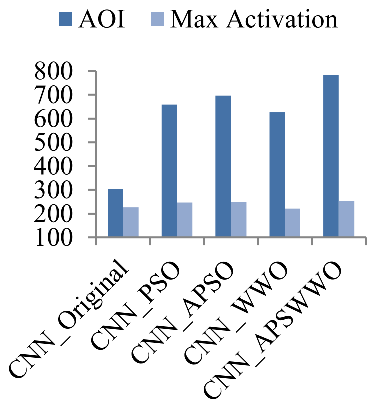
(d) max_pooling2d_1



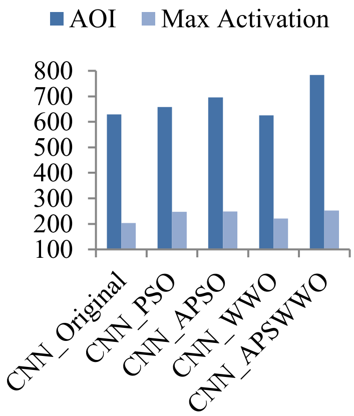
(e) conv2d_2



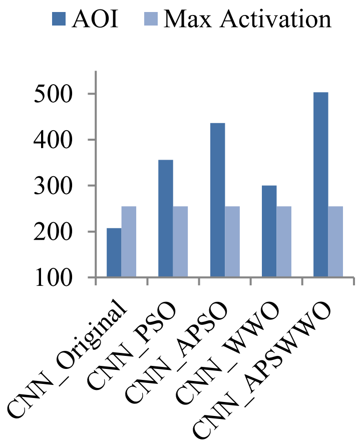
(f) attention



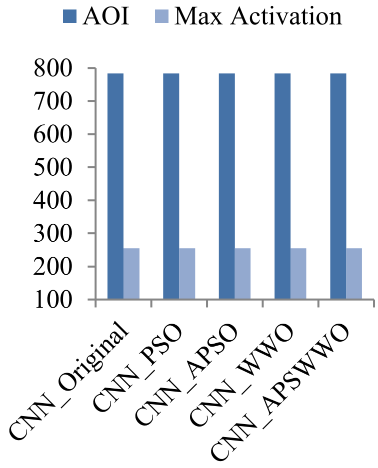
(g) operators.add



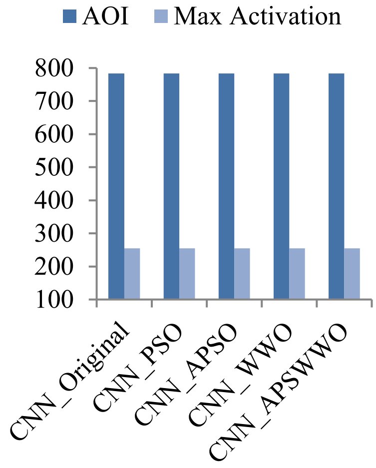
(h) attention_1



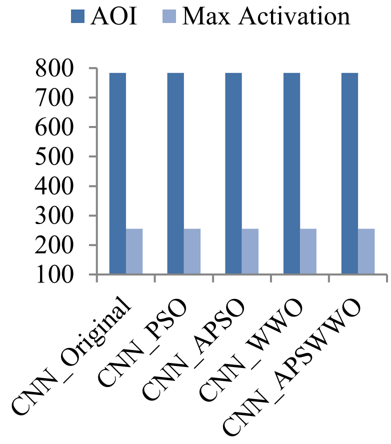
(i) operators.add_1



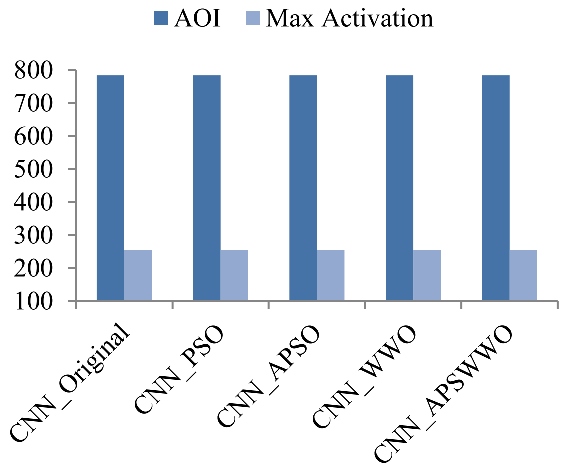
(j) global_average_pooling2d



(k) dense



(l) dropout



(m) dense_1

Figure 4.74: Visualization of Grad-CAM values of LiteViT

4.4.3 Comparative Analysis between proposed CNN and LiteViT Models

This study focuses on performance analysis, examining both 26-layer CNN models (CNN_Original, CNN_PSO, CNN_APSO, CNN_WWO, and CNN_APSWWO) and 13-layer LiteViT models (LiteViT_Original, LiteViT_PSO, LiteViT_APSO, LiteViT_WWO, and LiteViT_APSWWO). Among these models, CNN_APSWWO in the 26-layer CNN category and LiteViT_APSWWO in the 13-layer LiteViT category exhibit the highest performance. Specifically, CNN_APSWWO achieves a recall of 93.82%, precision of 94.09%, and an F1-score of 93.91%. It maintains a low training loss of 0.0019, testing loss of 0.4116, a high ROC-AUC score of 0.996, AP value of 0.975, an error rate of 5%, and an impressive accuracy of 95%. In contrast, LiteViT_APSWWO attains a recall of 86.55%, precision of 87%, and an F1-score of 86.64%. It maintains a training loss of 0.1097, testing loss of 0.5026, a ROC-AUC score of 0.985, AP value of 0.918, an error rate of 14%, and an accuracy of 86%. The CNN_APSWWO shows highest performance than LiteViT. Table 4.27 and Figure 4.75 represent the comparative analysis between CNN_APSWWO and LiteViT_APSWWO.

Table 4.27: Comparative analysis between CNN and LiteViT

Techniques	Recall (%)	Precision (%)	F1-score (%)	Training Loss	Testing Loss	ROC-AUC score	AP Value	Error Rate (%)	Accuracy (%)
CNN_APSWWO	93.82	94.09	93.91	0.0019	0.4116	0.996	0.975	5	95
LiteViT_APSWWO	86.55	87	86.64	0.1097	0.5026	0.985	0.918	14	86

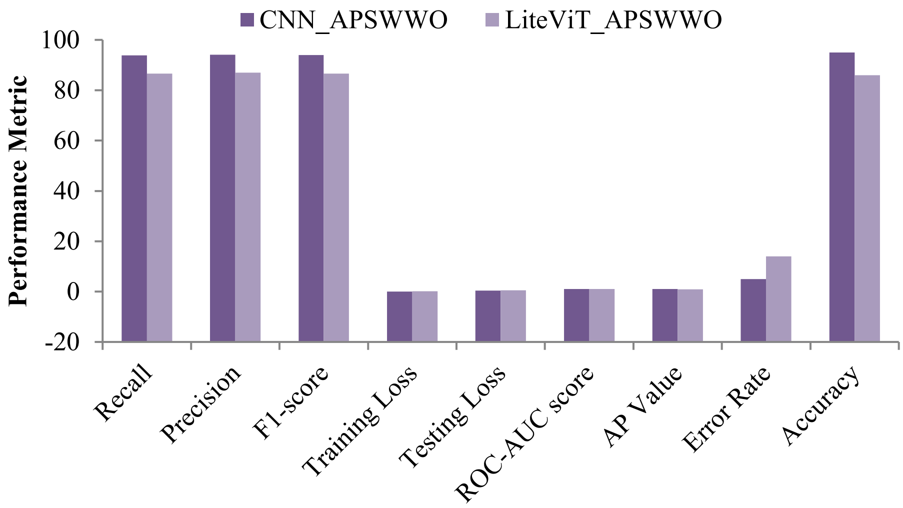


Figure 4.75: Comparative analysis between CNN and LiteViT

4.5 Discussion

This section evaluates the performance of our proposed model against existing methodologies in the field. Additionally, it explores the potential economic implications of our research findings.

4.5.1 Comparative analysis with existing works

This research introduces significant deviations from previous studies in the same domain.

Many studies [22-26] used pretrained transfer learning models in paddy disease classification, but relying solely on these models has limitations. Pretrained models are not always tailored to specific disease recognition tasks in the agricultural domain, and this can affect their performance. Additionally, transfer learning may not fully capture the intricacies of paddy disease characteristics. To address these limitations and enhance the effectiveness of disease classification, this study employed custom-designed 26-layer CNN and 13-layer LiteViT models.

Most studies of existing works [17-27] do not use any optimization algorithms in the paddy disease classification process. This study has used a novel hybrid optimization algorithm to optimize 26-layer CNN model hypermeters to classify paddy disease effectively.

A few studies [28] use grid search, bayesian, and random optimization techniques to classify paddy leaf disease. These techniques explore all combinations of hyperparameter values, significantly increasing the required training time and computational resources. These may suffer from the curse of dimensionality, particularly when the search space is high-dimensional. Additionally, grid search lacks adaptability and does not consider any feedback or information gained during the search process. It treats each combination equally without leveraging insights from earlier evaluations. Consequently, it may miss out on more promising regions of the hyperparameter space, limiting the potential for further optimization. So, this study focused on nature-inspired optimization techniques.

Some authors [16, 28-31] have employed single optimization techniques like GA, PSO, and BB-BC, but hybrid optimization algorithms are often favored for their ability to enhance exploration and adaptation, handle complexity, and perform across various objectives. Single optimization methods can suffer from convergence to local optima, limited exploration, and sensitivity to initialization or parameter settings. In contrast, hybrid optimization, leveraging the strengths of multiple algorithms, balances exploration and exploitation, adapts to dynamic problem scenarios, and increases the chances of finding optimal solutions. Hence, this study introduces a hybrid APSWWO approach to optimize the parameters of the 26-layer CNN and 13-layer LiteViT models.

While some research [16, 28, 29] has utilized optimization algorithms to fine-tune various hyperparameters in ANNs, including batch size, epoch number, learning rate, and dropout rate, there is a notable gap in optimizing more advanced layers in CNN and LiteViT models, particularly those involving attention mechanisms. To address this, our study takes the initiative to apply optimization algorithms specifically to the dropout rates of the attention and attention_1 layers in the model.

Many studies [30, 31] have primarily concentrated on the diseased class while overlooking the importance of considering the healthy class. In contrast, this study takes a comprehensive approach by focusing on ten distinct disease classes and incorporating one healthy class to facilitate a more balanced and inclusive evaluation of paddy disease detection. This broader scope allows for the effective classification of both diseased and healthy leaf samples, contributing to a more holistic understanding of the problem at hand.

While certain studies [20, 21, 28] have achieved commendable accuracy in their models, it is worth noting that a number of them have omitted the implementation of overfitting reduction techniques. In contrast, this study places a strong emphasis on model generalization and robustness by utilizing early stopping and optimizing dropout rates to effectively mitigate the risk of overfitting. These strategies not only enhance the model's performance but also ensure its applicability to unseen data, contributing to more reliable and dependable results.

While many studies [17, 21-24] have concentrated solely on a single performance metric, this research takes a comprehensive approach, evaluating multiple performance indicators. Specifically, our analysis extends to accuracy, error rate, precision, recall, F1-score, ROC curve, precision-recall curve, and learning curve. This thorough examination allows for a meticulous comparative analysis among the models, providing a more holistic understanding of their performance across various dimensions.

To date, no study has undertaken a comparative performance analysis encompassing the original, single, and hybrid forms of CNN models in paddy disease classification. This research addresses this gap by meticulously assessing the performance of five variations of a 26-layer CNN (CNN_Original, CNN_PSO, CNN_APSO, CNN_WWO, and CNN_APSWWO) and 13-layer LiteViT models (LiteViT_Original, LiteViT_PSO, LiteViT_APSO, LiteViT_WWO, and LiteViT_APSWWO). This extensive analysis sheds light on the nuanced performance distinctions among these models.

The majority of existing studies have traditionally paid limited attention to model interpretability techniques. In contrast, this research places a significant emphasis on model interpretability by dxxxxxseploying Grad-CAM techniques across all layers of the five distinct 26-layer CNN variants. This study not only emphasizes model interpretability but goes a step further by calculating Area of Interest and maximum activation for each layer in the five variants of the 26-layer CNN and 13-layer LiteViT (LiteViT_Original, LiteViT_PSO, LiteViT_APSO, LiteViT_WWO, and LiteViT_APSWWO) models. In addition to this, the research conducts an extensive comparative analysis based on Grad-CAM visualizations, enabling an in-depth exploration of how these models perform at various architectural levels. This multifaceted approach significantly enhances the transparency, accountability, and decision-making processes within these models.

To evaluate the effectiveness of our proposed system, we conducted a comparative analysis by comparing the highest accuracy achieved by CNN_APSWWO with the results obtained in previous studies. Table 4.28 illustrates the comparative performance analysis, demonstrating that CNN_APSWWO surpasses the results of the existing works, indicating its superior performance.

Table 4.28: Comparative analysis with existing works

	Precision (%)	**Recall (%)**	**F1-score (%)**	**Accuracy (%)**
Proposed Model (CNN_APSWWO)	93.82	94.09	93.91	95
Rahman et al. [17]	-	-	-	93.3
Wang et al. [28]	92.6	87.4	89.6	94.65
Hasan et al. [26]	92.15	90.56	90.56	92.86
Tejaswini et al. [19]	-	-	-	78.2
Haque et al. [40]	90	67	81	-
Rahman et al. [41]	-	-	-	90
Kumar et al. [42]	82.8	75.81	92.45	94.87
Daniya et al. [43]		86.2	90.7	90.8
Dogra et al. [44]	92.4	89.9	90.5	93

4.5.2 Impact of Proposed Work on Economy

The impact of the proposed work on the economy is significant. By advancing the accuracy and robustness of paddy disease detection models, this research contributes to improving crop health and reducing yield losses. The accurate identification and classification of paddy diseases can lead to early intervention and precise treatment, resulting in higher crop yields and improved food security. This, in turn, can have a positive economic impact by increasing agricultural productivity and reducing losses due to diseases. Moreover, the research findings can lead to the development of more efficient and targeted agricultural practices, minimizing the need for excessive pesticide use and associated costs. Ultimately, the economic implications of this work are substantial, as it directly addresses the challenges faced by the agricultural sector, which is a vital component of the global economy.

4.6 Summary

This research addresses the urgency of paddy disease detection for global food security, introducing a novel hybrid optimization algorithm, APSWWO, for optimizing Artificial Neural Network parameters. Evaluating 26-layer CNN and 13-layer LiteViT models, CNN_APSWWO achieves 95% accuracy, outperforming other CNN models and LiteViT_APSWWO. The study emphasizes the superiority of hybrid APSWWO in correctly classifying instances, minimizing misclassifications, and improving recall, precision, and F1-score. Visualizing decision-making processes with Grad-CAM, CNN_APSWWO consistently outperforms in various layers, showcasing effective learning and superior accuracy. This research significantly contributes to paddy disease recognition, providing insights for optimizing detection and enhancing global food security.

CHAPTER 5

CONCLUSION

5.1 Conclusion

This research addresses the critical issue of paddy diseases that significantly impact global food security and productivity. The economic losses and annual damages incurred emphasize the urgency of developing effective detection methods. Though ANN provides an effective solution in paddy disease recognition, the research recognizes the challenges in optimizing ANN parameters and argues for the superiority of hybrid optimization algorithms over single optimization techniques. Therefore, this study introduces a novel hybrid optimization algorithm, APSWWO, for optimizing the parameters of Artificial Neural Network in paddy disease detection. The 26-layer CNN and 13-layer LiteViT models are utilized for classification, providing a sophisticated and efficient approach compared to manual methods and traditional disease mapping techniques. The study evaluates the performance of different optimization techniques (Original, PSO, APSO, WWO, and APSWWO) applied to CNN and LiteViT to classify paddy diseases.

The CNN_ original achieves 88% accuracy, with subsequent improvements using Particle Swarm Optimization, Adaptive Particle Swarm Optimization, Water Wave Optimization, and the proposed hybrid Adaptive Particle Swarm Water Wave Optimization. Among these, CNN_APSWWO stands out, achieving a remarkable 95% accuracy.

The comparative analysis of five LiteViT models (LiteViT_Original, LiteViT_PSO, LiteViT_APSO, LiteViT_WWO, and LiteViT_APSWWO) reveals that LiteViT_APSWWO consistently outperforms others in classifying paddy diseases. It exhibits the highest correctly classified instances, lowest misclassified instances, and superior recall, precision, and F1-score. LiteViT_APSWWO maintains the lowest training and testing losses, highest ROC-AUC score, and tops in Average Precision value. With the lowest error rate and highest accuracy at 14% and 86%, respectively, LiteViT_APSWWO emerges as the top-performing model across all five LiteViT models.

Among the models evaluated, CNN_APSWWO in the 26-layer CNN category and LiteViT_APSWWO in the 13-layer LiteViT category demonstrate the best performance. CNN_APSWWO achieves a recall of 93.82%, precision of 94.09%, and an F1-score of 93.91%, with low training and testing losses, high ROC-AUC score, and impressive accuracy. LiteViT_APSWWO, while performing well, shows slightly lower recall, precision, and F1-score compared to CNN_APSWWO. Overall, CNN_APSWWO outperforms LiteViT_APSWWO in the evaluation. This hybrid approach outperforms others in correctly classifying instances, minimizing misclassifications, and improving recall, precision, and F1-score. The comparison emphasizes the significance of hybrid APSWWO in enhancing the prediction performance of the CNN model, contributing valuable insights for optimizing paddy disease recognition and classification.

The study also employs Grad-CAM in all layers of CNN_Original, CNN_PSO, CNN_APSO, CNN_WWO, CNN_APSWWO, LiteViT_Original, LiteViT_PSO, LiteViT_APSO, LiteViT_WWO, and LiteViT_APSWWO for model interpretability, providing visualizations and insights into each layer's decision-making processes. In various layers such as conv2d, batch_normalization, conv2d_1, batch_normalization_1, conv2d_2, batch_normalization_2, conv2d_3, batch_normalization_3, math.multifly, attention, math.multifly_1, conv2d_4, and batch_normalization_4, CNN_APSWWO consistently outperforms other techniques, demonstrating higher AOI and Maximum Activation values. This superior accuracy in all layers suggests that CNN_APSWWO correctly classifies instances, showcasing its effective learning of crucial features and patterns in the data. The enhanced accuracy contributes to better AOI and Maximum Activation values, highlighting the model's ability to focus on and extract relevant information. Overall, this research contributes significantly to paddy disease recognition, offering a robust hybrid optimization algorithm and an advanced CNN model. The findings provide valuable insights for optimizing paddy disease detection and classification, ultimately contributing to improved agricultural practices, crop health, and global food security.

5.2 Future Scope

As future work, this study envisions applying the proposed methodology in classifying other plant diseases, such as potato, maize, and corn disease detection.

References

1. Feng, Lei, Baohua Wu, Susu Zhu, Junmin Wang, Zhenzhu Su, Fei Liu, Yong He, and Chu Zhang. "Investigation on data fusion of multisource spectral data for rice leaf diseases identification using machine learning methods." Frontiers in plant science 11 (2020): 577063.
2. Daniya, T. and Vigneshwari, S., 2022. Deep neural network for disease detection in rice plant using the texture and deep features. *The Computer Journal*, *65*(7), pp.1812-1825.
3. Liu, H., Cui, Y., Wang, J. and Yu, H., 2023. Analysis and Research on Rice Disease Identification Method Based on Deep Learning. *Sustainability*, *15*(12), p.9321.
4. Liang, W.J., Zhang, H., Zhang, G.F. and Cao, H.X., 2019. Rice blast disease recognition using a deep convolutional neural network. *Scientific reports*, *9*(1), p.2869.
5. Dubey, R.K. and Choubey, D.K., 2024. An efficient adaptive feature selection with deep learning model-based paddy plant leaf disease classification. Multimedia Tools and Applications, 83(8), pp.22639-22661.
6. Gianessi, L.P., 2014. Importance of pesticides for growing rice in South and South East Asia. *International pesticide benefit case study*, *108*, pp.30-33.
7. Singh, P., Mazumdar, P., Harikrishna, J.A. and Babu, S., 2019. Sheath blight of rice: a review and identification of priorities for future research. *Planta*, *250*, pp.1387-1407.
8. Liu, L.M., Zhao, Y., Zhang, Y.L., Wang, L., Hou, Y.X. and Huang, S.W., 2020. First report of leaf spot disease on rice caused by Epicoccum sorghinum in China. *Plant Disease*, *104*(10), pp.2735-2735.
9. Zarbafi, S.S. and Ham, J.H., 2019. An overview of rice QTLs associated with disease resistance to three major rice diseases: blast, sheath blight, and bacterial panicle blight. *Agronomy*, *9*(4), p.177.
10. Han, X., Yang, Y., Wang, X., Zhou, J., Zhang, W., Yu, C., Cheng, C., Cheng, Y., Yan, C. and Chen, J., 2014. Quantitative trait loci mapping for bacterial blight resistance in rice using bulked segregant analysis. *International Journal of Molecular Sciences*, *15*(7), pp.11847-11861.
11. Guo, S., Zhang, D. and Lin, X., 2010. Identification and mapping of a novel bacterial blight resistance gene Xa35 (t) originated from Oryza minuta. *Scientia Agricultura Sinica*, *43*(13), pp.2611-2618.
12. Abiodun, O.I., Jantan, A., Omolara, A.E., Dada, K.V., Mohamed, N.A. and Arshad, H., 2018. State-of-the-art in artificial neural network applications: A survey. *Heliyon*, *4*(11).
13. Wu, Y.C. and Feng, J.W., 2018. Development and application of artificial neural network. *Wireless Personal Communications*, *102*, pp.1645-1656.
14. Kalyan, K., Jakhia, B., Lele, R.D., Joshi, M. and Chowdhary, A., 2014. Artificial neural network application in the diagnosis of disease conditions with liver ultrasound images. *Advances in bioinformatics*, *2014*.

15. Moayedi, H., Mosallanezhad, M., Rashid, A.S.A., Jusoh, W.A.W. and Muazu, M.A., 2020. A systematic review and meta-analysis of artificial neural network application in geotechnical engineering: theory and applications. *Neural Computing and Applications*, *32*, pp.495-518.
16. Sharma, R. and Singh, A., 2022. Big bang–big crunch-CNN: an optimized approach towards rice crop protection and disease detection. *Archives of Phytopathology and Plant Protection*, *55*(2), pp.143-161.
17. Rahman, C.R., Arko, P.S., Ali, M.E., Khan, M.A.I., Apon, S.H., Nowrin, F. and Wasif, A., 2020. Identification and recognition of rice diseases and pests using convolutional neural networks. *Biosystems Engineering*, *194*, pp.112-120.
18. Latif, G., Abdelhamid, S.E., Mallouhy, R.E., Alghazo, J. and Kazimi, Z.A., 2022. Deep learning utilization in agriculture: Detection of rice plant diseases using an improved CNN model. *Plants*, *11*(17), p.2230.
19. Tejaswini, P., Singh, P., Ramchandani, M., Rathore, Y.K. and Janghel, R.R., 2022, June. Rice leaf disease classification using CNN. In *IOP Conference Series: Earth and Environmental Science* (Vol. 1032, No. 1, p. 012017). IOP Publishing.
20. Sethy, P.K., Barpanda, N.K. and Rath, A.K., 2019. Detection and identification of rice leaf diseases using multiclass SVM and particle swarm optimization technique. *Int. J. Innovative Tech. and Exploring Eng.(IJITEE)*, *8*(6S2), pp.108-120.
21. Jiang, M., Feng, C., Fang, X., Huang, Q., Zhang, C. and Shi, X., 2023. Rice Disease Identification Method Based on Attention Mechanism and Deep Dense Network. Electronics, 12(3), p.508.
22. Krishnamoorthy, N., Prasad, L.N., Kumar, C.P., Subedi, B., Abraha, H.B. and Sathishkumar, V.E., 2021. Rice leaf diseases prediction using deep neural networks with transfer learning. *Environmental Research*, *198*, p.111275.
23. Chen, J., Zhang, D., Nanehkaran, Y.A. and Li, D., 2020. Detection of rice plant diseases based on deep transfer learning. *Journal of the Science of Food and Agriculture*, *100*(7), pp.3246-3256.
24. Matin, M.M.H., Khatun, A., Moazzam, M.G. and Uddin, M.S., 2020. An efficient disease detection technique of rice leaf using AlexNet. *Journal of Computer and Communications*, *8*(12), pp.49-57.
25. Kathiresan, G., Anirudh, M., Nagharjun, M. and Karthik, R., 2021, May. Disease detection in rice leaves using transfer learning techniques. In *Journal of Physics: Conference Series* (Vol. 1911, No. 1, p. 012004). IOP Publishing.
26. Hasan, M.M., Uddin, A.S., Akhond, M.R., Uddin, M.J., Hossain, M.A. and Hossain, M.A., 2023. Machine Learning and Image Processing Techniques for Rice Disease Detection: A Critical Analysis. *International Journal of Plant Biology*, *14*(4), pp.1190-1207.
27. Bari, Bifta Sama, Md Nahidul Islam, Mamunur Rashid, Md Jahid Hasan, Mohd Azraai Mohd Razman, Rabiu Muazu Musa, Ahmad Fakhri Ab Nasir, and Anwar PP Abdul Majeed. "A real-time approach of diagnosing rice leaf disease using deep learning-based faster R-CNN framework." PeerJ Computer Science 7 (2021): e432.

28. Wang, X., Wang, Y., Zhao, J. and Niu, J., 2023. ECA-ConvNeXt: A Rice Leaf Disease Identification Model Based on ConvNeXt. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (pp. 6234-6242).
29. Ramesh, S. and Vydeki, D., 2020. Recognition and classification of paddy leaf diseases using Optimized Deep Neural network with Jaya algorithm. *Information processing in agriculture*, *7*(2), pp.249-260.
30. Gayathri Devi, T., Rajkumar, G., Srinivasan, A. and Sudha, S., 2022. Radial basis function neural network and salp swarm algorithm for paddy leaf diseases classification in Thanjavur, Tamilnadu geographical region. *Acta Geophysica*, *70*(6), pp.2917-2932.
31. Raja, M.R., Jayaraj, V., Shajin, F.H. and Devi, E.R., 2023. Radial basis function Neural Network optimized with Salp Swarm algorithm espoused paddy leaf disease classification. *Biomedical Signal Processing and Control*, *86*, p.105038.
32. Prity, F.S., Nath, N., Nath, A. and Uddin, K.A., 2023. Neural network-based strategies for automatically diagnosing of COVID-19 from X-ray images utilizing different feature extraction algorithms. Network Modeling Analysis in Health Informatics and Bioinformatics, 12(1), p.28.
33. Jeatrakul, P., Wong, K.W. and Fung, C.C., 2010. Classification of imbalanced data by combining the complementary neural network and SMOTE algorithm. In *Neural Information Processing. Models and Applications: 17th International Conference, ICONIP 2010, Sydney, Australia, November 22-25, 2010, Proceedings, Part II 17* (pp. 152-159). Springer Berlin Heidelberg.
34. Wang, S., Dai, Y., Shen, J. and Xuan, J., 2021. Research on expansion and classification of imbalanced data based on SMOTE algorithm. *Scientific reports*, *11*(1), p.24039.
35. Xu, Z., Shen, D., Nie, T. and Kou, Y., 2020. A hybrid sampling algorithm combining M-SMOTE and ENN based on Random forest for medical imbalanced data. *Journal of Biomedical Informatics*, *107*, p.103465.
36. Liang, X.W., Jiang, A.P., Li, T., Xue, Y.Y. and Wang, G.T., 2020. LR-SMOTE—An improved unbalanced data set oversampling based on K-means and SVM. *Knowledge-Based Systems*, *196*, p.105845.
37. Yamashita, R., Nishio, M., Do, R.K.G. and Togashi, K., 2018. Convolutional neural networks: an overview and application in radiology. *Insights into imaging*, *9*, pp.611-629.
38. Kalchbrenner, N., Grefenstette, E. and Blunsom, P., 2014. A convolutional neural network for modelling sentences. *arXiv preprint arXiv:1404.2188*.
39. Huang, T., Huang, L., You, S., Wang, F., Qian, C. and Xu, C., 2022. Lightvit: Towards light-weight convolution-free vision transformers. arXiv preprint arXiv:2207.05557.
40. Haque, M.E., Rahman, A., Junaeid, I., Hoque, S.U. and Paul, M., 2022. Rice leaf disease classification and detection using yolov5. *arXiv preprint arXiv:2209.01579*.
41. Asfaqur Rahman, M., Shahriar Nawal Shoumik, M., Mahbubur Rahman, M. and Hasna Hena, M., 2021. Rice disease detection based on image processing technique. In *Smart Trends in Computing and Communications: Proceedings of SmartCom 2020* (pp. 135-145). Springer Singapore.
42. Kumar, V.S., Jaganathan, M., Viswanathan, A., Umamaheswari, M. and Vignesh, J., 2023. Rice leaf disease detection based on bidirectional feature attention pyramid network with YOLO v5 model. *Environmental Research Communications*.

43. Daniya, T. and Vigneshwari, S., 2023. Rider Water Wave-enabled deep learning for disease detection in rice plant. *Advances in Engineering Software*, *182*, p.103472.
44. Dogra, R., Rani, S., Singh, A., Albahar, M.A., Barrera, A.E. and Alkhayyat, A., 2023. Deep learning model for detection of brown spot rice leaf disease with smart agriculture. *Computers and Electrical Engineering*, *109*, p.108659