"Crops Leaf Disease Recognition From Digital And RS Imaging Using Fusion of Multi Self-Attention Rbnet Deep Architectures and Modified Dragonfly Optimization"

A

Dissertation Report

Master of Technology

in

Web Technologies

SUBMITTED BY:

Somashekhar Naik K

3rd sem, MTech CSE (WT)

P25UV23T113011

UNDER THE GUIDANCE OF

Dr. Pushpa C N

Associate Professor

Dept. of CSE, UVCE

UNIVERSITY OF VISVESVARAYA COLLEGE OF ENGINEERING

K.R. CIRCLE, BANGALORE-560001

YEAR 2024-2025

UNIVERSITY OF VISVESVARAYA COLLEGE OF ENGINEERING
K.R. CIRCLE, BANGALORE-560001

Department of Computer Science and Engineering

CERTIFICATE

This is certified that the seminar work entitled “**Crops Leaf Disease Recognition From Digital And RS Imaging Using Fusion of Multi Self-Attention Rbnet Deep Architectures and Modified Dragonfly Optimization**” is carried out by **SOMASHEKHAR NAIK K (P25UV23T113011)** bonafide student of the Department of CSE, in partial fulfillment for the award of **Master of Technology** in **Web Technologies** of the **University of Visvesvaraya College of Engineering**, Bengaluru during the year 2025 - 2026. The seminar report has been approved as it satisfies the academic requirements in respect of III Semester Dissertation Phase-I work prescribed for the Master of Technology degree.

GUIDE:	CHAIRPERSON:
______________________	______________________
Dr. Pushpa C N **Associate Professor** **Department of CSE,** **UVCE**	**Dr. THRIVENI J** **Professor** **Department of CSE,** **UVCE**
Examiner 1:______________	**Examiner 2:** ______________

ACKNOLEDGEMENT

The satisfaction that accompanies the successful completion of the task would be put incomplete without the mention of the people who made it possible, whose constant guidance and encouragement crown all the efforts with success.

We express our gratitude to **Dr. Subhasish Tripathy**, Director, UVCE, who has been leading our college with a brighter vision in Technical Education.

We are very thankful to our guide **Dr. Pushpa C N,** Department of Computer Science Engineering for his sustained inspiring guidance and cooperation throughout the process of this project. His wise counsel and valuable suggestions are invaluable.

I would like to express our sincere thanks **to Mrs. Veena,** Research Scholar, Department of Computer Science and Engineering, UVCE, for her inspiration and encouragement.

I avail this opportunity to express my deep sense of gratitude and hearty thanks to the Management of **University of Visvesvaraya College of Engineering**, for providing world class infrastructure, congenial atmosphere, and encouragement.

I express my deep sense of gratitude and thanks to the teaching and non-teaching staff at our department who stood with me during the project and helped me to make it a successful venture.

Submitted By

SOMASHEKHAR NAIK K

ABSTRACT

Globally, pests and plant diseases severely threaten forestry and agriculture. Plant protection could be substantially enhanced by using noncontact, extremely effective, and reasonably priced techniques for identifying and tracking pests and plant diseases across large geographic areas. Precision agriculture is the study of using other technologies, such as hyperspectral remote sensing, to increase cultivation instead of traditional agricultural methods with less negative environmental effects. In this article, we proposed a novel deep-learning architecture and optimization algorithm for crop leaf disease recognition. In the initial step, a multilevel contrast enhancement technique is proposed for a better visual of the disease on the leaves of cotton and wheat. After that, we proposed three novel residual block and self-attention mechanisms, named 3-residual block-deep convolutional neural network (RBNet) Self, 5-RBNet Self, and 9-RBNet Self. After that, the proposed models are trained on enhanced images and later extracted deep features from the self-attention layer. The 5-RBNET Self and 9-RBNET Self performed well in terms of accuracy and precision rate; therefore, we did not consider the 3-RBNET Self for the next process. The dragonfly optimization algorithm is proposed for the best feature selection and applied to the self-attention features of 5-RBNET Self and 9-RBNET Self models to improve the classification performance further and reduce the computational cost. The proposed method is evaluated on two publically available crop disease images, such as the cotton, wheat, and EuroSAT datasets. For both crops, the proposed method obtained a maximum accuracy of 98.60% and 93.90%, respectively, whereas for the EuroSAT, the proposed method obtained an accuracy of 83.10%. Compared to the results with recent techniques, the proposed method shows improved accuracy and precision rate

Table of Contents

List of Figures

CHAPTER 1

INTRODUCTION

The yield of any crop in reasonable quantity and quality is essential for any country to become economically stable Food is a necessity of human life, so there must be no gap between the supply and demand of food, which is only possible by growing sufficient amounts of crops, especially those that are used abundantly, directly or indirectly, like wheat, rice, corn, cotton, and vegetables because they are needed daily. Food shortage is observed due to the attack of different pests and diseases on crops, bad weather conditions, and the timely detection and eradication of disease in plantsIn contrast, pests and disease management systems have increased the yield of food production for the last 40 years Global estimations show an annual loss of about 50% in wheat crop yield and up to 80% in cotton worldwide By looking into these facts, it is clear that there is a need to stop these pests and disease attacks that are destroying crops. It is done mainly by using manual or traditional methods that involve the observation of visual symptoms or are based on crop knowledge, identifying the disease, using a proper pesticide, or taking precautionsBut these methods need a lot of time and proper as well as precise knowledge, which in most cases is trivial, so this system must be automated to make it less laborious and economically feasible for the farmers [9], [10]. Since manual classification is dependent on human knowledge, it can add biasness in the results and leads to the problem of misclassification

In recent studies, several researchers used optimization techniques for the selection of the best features. The main purpose of this step was to consider only important information, such as image features that can accurately recognize the image label. Optimization algorithms are playing a vital role in ML for various tasks which include selection of useful features, tuning of parameters and training of ML models. The important one, i.e., feature selection, involves the selection of the most relevant feature from the available data, which helps overcome the problem of overfitting and building an optimized lightweight ML model. There are several feature optimization techniques, such as PSO, WOA , Crow search , Lion optimization , and a few more. The selected features are finally classified using neural network classifiers

Two of the most common cash crops worldwide are wheat and cotton, which are also very vulnerable to pests and disease attacks due to their abundance. Common diseases found are powdery mildew, tan spot, leaf rust, stripe rust, stagonospora leaves, fusarium, bacterial leaf streak, wheat streak, mosaic virus and many more. Similarly, common diseases found in cotton are leaf spots, bacterial blight, wilts, fusarium wilt, leaf curl virus, angular leaf spot, and others . It shows that many diseases attack these two important crops, and acquiring the data of their leaves is also very challenging, which causes noise, occlusion, background environment effects, and weather conditions thatthe flattened layers, such as an average pool or fully connected. A flattened layer is used to convert a multidimensional array or matrix in the case of an image to a one-dimensional (1-D) matrix. In image processing, it is applied after convolution layer to transform the extracted information from spatial domain to one, which is suitable for fully connected layer as an input . A fully connected layer, also known as a dense layer, is very important and purposeful in a CNN. This layer is mapped at the end of a CNN network to adjust the extracted features obtained through convolution and pooling layers according to their probabilities. Spatial information obtained through preceding layers is aggregated into an output feature vector, which is used for class prediction . The extracted features are in the form of edges, shape, texture, color intensities, and other distinguishing properties

CHAPTER 2

LITERATURE SURVEY

In recent studies, several researchers used optimization techniques for the selection of the best features. The main purpose of this step was to consider only important information, such as image features that can accurately recognize the image label. Optimization algorithms are playing a vital role in ML for various tasks which include selection of useful features, tuning of parameters and training of ML models. The important one, i.e., feature selection, involves the selection of the most relevant feature from the available data, which helps overcome the problem of overfitting and building an optimized lightweight ML model. There are several feature optimization techniques, such as PSO, WOA , Crow search , Lion optimization, and a few more. The selected features are finally classified using neural network classifiers

Shafi presented an architecture of wheat crop disease classification that involved the use of RS devices and ML models and trained the concept of the Internet of Things. It was proposed that data acquisition be done manually through hand-held devices and also by using satellite and unmanned aerial vehicle imagery. In the next step, the data are cleaned, transformed, and normalized according to the required standards, and later on, an ML-based algorithm is used to train and test the proposed architecture. In the stage of image preprocessing, the following functions were performed: noise removal sharpening of the edges to do so, low-pass filters were used for noise removal, and high-pass filters were used for sharpening of the edges. Different DCNNs were evaluated on the acquired and preprocessed data, namely, VGG-16 [49], VGG-19 [50], AlexNet [51], DenseNet [52], and GoogleNet [53]. After experimentation, it was concluded that the performance of the aforementioned models depends upon many factors such as the quality of the dataset, acquired, devices used for capturing images, methodology, and size of the dataset

Xu et alsuggested an integrated DL framework with a residual channel attention block, a feedback block, an elliptic metric learning (EML), and a CNN model. The authors used two CNN models in parallel to extract the basic features which separate healthy and diseased

wheat leaves. The residual block was utilized to optimize the extracted features. Feedback blockmake the acquired dataset less visible and suitable for the process of computer vision (CV) and machine learning (ML). After the implementation of these CV-based automated early disease detection systems, the growth of crops is greatly affected. Automated disease classification is beneficial for early diagnosis and precisely suggests the methodology to tackle the respective disease. Basic computer vision-based techniques involve dataset acquisition, preprocessing to enhance the quality of sample images, extraction of feature vectors, feature optimization for the extraction of useful features, and classification using several classifiers

The leaf of any crop is the most prominent and visible part, which can be acquired and analyzed easily, other than the stem and root. Alongside CV-based methods, the entrance of deep learning (DL) shows much success in disease detection and recognition using digital and remote sensing (RS) imagesA few sample images are shown in Fig. 1. Deep convolutional neural networks (DCNN) are used to get better results than traditional techniques such as handcrafted features, feature reduction, and classifiers (SVM , KNN). In the area of DL, several pretrained models have been available for the classification of crop diseases, such as DarkNet-19, DarkNet-56, EfficientNet-B0, EfficientNet-B7, AlexNet, ResNet, and many more. These DCNNs extract deep features from the activation layers. The addition of convolutional neural networks (CNN) based classifiers, namely wide neural networks, medium neural networks, narrow neural networks, bilayered neural networks, and trilayered neural network has also added much improvement in DCNN’s architecture for classification accuracy

CHAPTER 3

ARCHITECTURE

Proposed framework for the classification of crops leaf diseases using deep learning and optimization

The proposed methodology for the classification of cotton and wheat leaf disease is presented in this section. In the proposed methodology, the wheat and cotton leaves datasets are collected from Kaggle. Both datasets are publicly available for research purposes. A multilevel contrast enhancement technique is proposed based on the fusion of local and top-bottom filtering mathematical formulation for better visual information of an image. Following that, three deep self-attention architectures have been proposed and trained on contrast-enhanced images. The features are extracted from the self-attention layer, and classification is performed using neural network classifiers. After that, an optimization algorithm named binary dragonfly was implemented, and the best features were selected. The selected features are again passed to neural networks and performed classification. To further improve the accuracy and precision rate, a Bayesian optimization algorithm is applied to neural network classifiers, and the hyperparameters are optimized

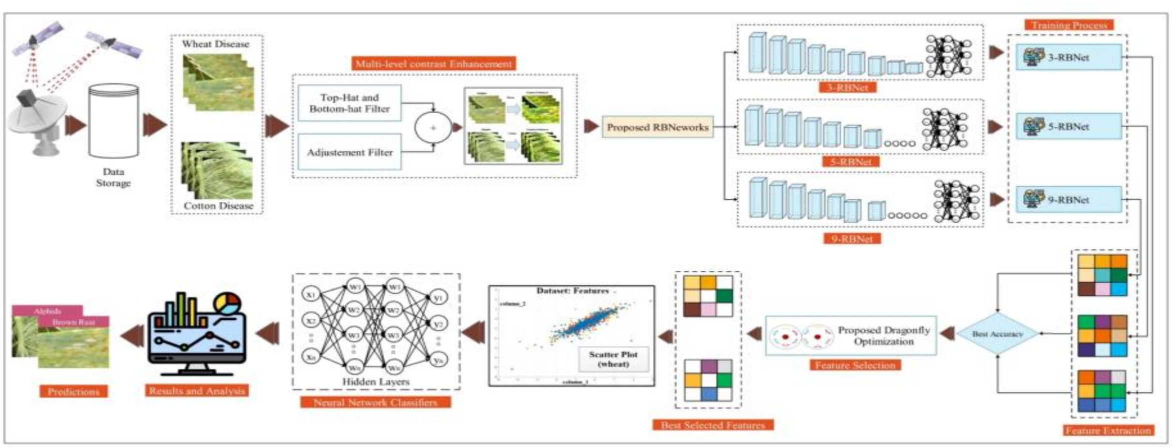


FIGURE 1. Proposed Frame work.

In this article, we proposed a novel deep-learning architecture for the classification of crop diseases using digital and RS images. a brightness preserving bi-histogram equalization [61] and dualistic subimage histogram equalization [62] techniques concepts have been considered and fused local adjustment filter with top-bottom transformation for the contrast enhancement. The fused technique is applied to both training and testing images. After that, self-attention DL models were proposed, and the extracted features were further optimized using the binary dragonfly algorithm for feature selection. In addition, Bayesian optimization is employed for finetuning hyperparameters of selected neural network classifiers. Hence, the core objective of this research is to build a lightweight CNN-based model that can be utilized for the training of crops as well as remotely acquired datasets. In addition, to overcome the problems faced by using the existing CNN architectures, i.e., by focusing on hybrid preprocessing techniques and building lightweight CNN models for deep features extraction so that nonredundant and useful features could be extracted without losing the actual information present in the sample images. The main contribution is briefly given as follows.

1) A multilevel contrast enhancement technique is proposed based on the fusion of local adjustment and top-bottom filtering for a better visual of the disease on the leaves of cotton and wheat crops.

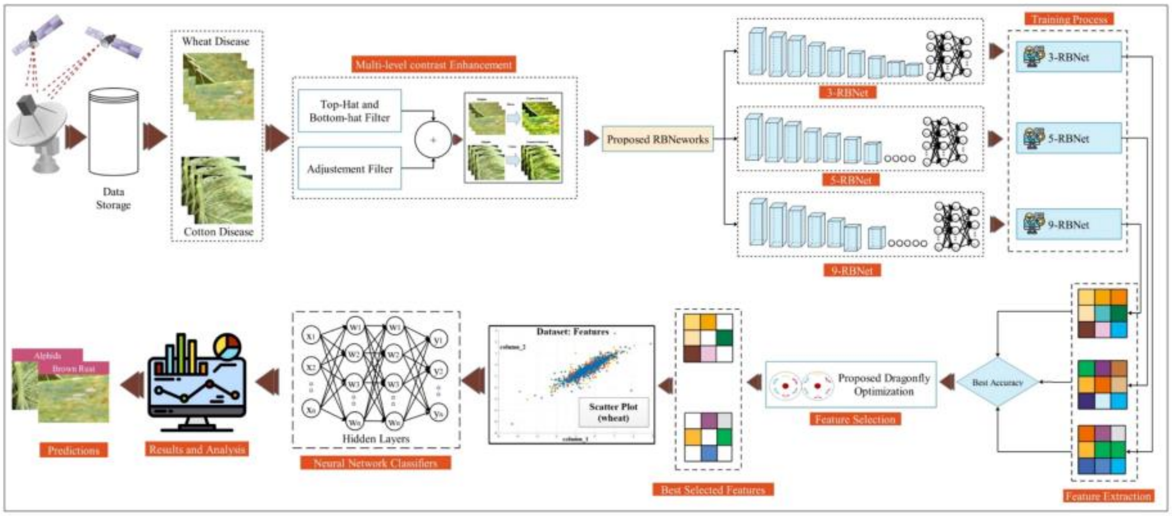
2) Three models are proposed based on residual block and self-attention mechanism. The models are named by 3- residual block-deep convolutional neural network (RBNet) Self, 5-RBNet Self, and 9-RBNet Self. The deep features are extracted from the self-attention layer for the classification process.

3) A binary dragonfly optimization is implemented for the best feature selection from the extracted features to reduce the testing time

4) Optimize the hyperparameters of selected neural network classifiers using Bayesian Optimization for improved accuracy and precision rate.

CHAPTER 4

PROPOSED METHODOLOGY



The proposed methodology for the classification of cotton and wheat leaf disease is presented in this section. In the proposed methodology, the wheat and cotton leaves datasets are collected from Kaggle. Both datasets are publicly available for research purposes. A multilevel contrast enhancement technique is proposed based on the fusion of local and top-bottom filtering mathematical formulation for better visual information of an image. Following that, three deep self-attention architectures have been proposed and trained on contrast-enhanced images. The features are extracted from the self-attention layer, and classification is performed using neural network classifiers. After that, an optimization algorithm named binary dragonfly was implemented, and the best features were selected. The selected features are again passed to neural networks and performed classification. To further improve the accuracy and precision rate, a Bayesian optimization algorithm is applied to neural network classifiers, and the hyperparameters are optimized

1) Dataset Collection: In this work, four publically available datasets have been utilized, namely wheat leaves,wheatdisease,cotton leaf disease and cotton plant diseaseTwo datasets are used for validation purposes, one belonging to wheat leave disease containing 4086 sample images and the other two datasets belonging to cotton leave disease comprising 4107 sample images; both the datasets contain RGB sample images. The wheat leaf dataset consists of three classes, namely healthy, septoria, and stripe rust. The wheat disease dataset contains three classes as well, namely, brown rust, healthy, and yellow rust classes. A few sample images of the dataset are shown in Fig. 3. Both datasets are combined for the experimental process, as described in Table I. The cotton plant disease contains six different classes, namely aphids, army word, bacterial blight, healthy, powdery mildew, and target spot, and the cotton leaf disease dataset has four different classes. The names of the classes are a bacterial blight, curl virus, fusarium, wilt, and healthy. The datasets on wheat diseases are combined, and the data on cotton diseases are also combined to collect different diseases and samples for the experimental process

Datasets	Name of class	No. of images
Wheat dataset		
1	Brown rust	1128
2	Healthy	1497
3	Septoria	97
4	Stripe rust	208
5	Yellow rust	1156
Cotton dataset		
1	Aphids	400
2	Army worm	400
3	Bacterial blight	847
4	Curl virus	417
5	Fusarium wilt	418
6	Healthy	825
7	Powdery mildew	400
8	Target spot	400

2) Proposed Contrast Enhancement: Contrast enhancement is one of the most vital objectives considered for imagepreprocessing before making it suitable for model training During contrast enhancement, the RoI or the overall contrast of the sample image is enhanced so that disease parts become prominent. In this research work, the datasets acquired have low image resolution and contrast qualities, so a good technique is required to enhance the images. A multilevel contrast filtering technique, which is a fusion of multiple filters mathematical formulation, is proposed. Initially, the top and bottom hat contrast enhancement filters are implemented and then fused their formulation to adjust the variance in colors by employing statistical parameters [65]. Suppose $\mathcal{D}$ is the selected dataset having N number of images represented as $\mathcal{D}\in$ RN, individual images is represented by Tn(p, v), where (p, v) R and every sample image is resized as A × B = 224. Assume that kernel denoted by is initialized with a valueThe top-hat filtration is based on (·) opening operation and bottom hat filter is based on (·) closing operation. The top hat and bottom hat contrast enhancement is mathematically defined as follows:

$$\mathrm{T}_{top}(p,v) = \mathrm{T}^n(p,v) - (\mathrm{T}^n(p,v) \cdot \text{ƅ}) \quad (1)$$

$$\mathrm{T}_{bottom}(p,v) = (\mathrm{N}^n(p,v) \blacksquare \text{ƅ}) - \mathrm{T}^n(p,v) \quad (2)$$

$$h'(p,v) = \mathrm{T}^m(p,v) + \mathrm{T}_{top}(p,v) - \mathrm{T}_{bottom}(p,v) \quad (3)$$

where h (p, v) denoted the resultant image of top-hat and bottom-hat methods. Following that, an adjustment filter is applied to enhance the lightening of an image by transforming the pixel values of input intensity values accordingly by setting the mean values of low and high intensities to about 1.5%. Mathematically, it is defined as follows :

$$adj^n(p,v) = \left(\frac{p-l}{s-l}\right)^{\gamma} (t-h) + t \quad (4)$$

$$\mathrm{F}^n(p,v) = h'(p,v) + adj^n(p,v) \quad (5)$$

where n(p, v) is the final image and gamma (γ) represents the correlation between coordinating coefficients like (l, h) and (s, t), denote the pixel intensity values in an image, and Fn(p, v) is the resultant output-enhanced image

B. Proposed Residual Networks

A residual block is an architectural block that has a skip connection alongside the regular feed-forward technique. Allowing the network to take absorbed, residual functions is the basic tenet of a residual block. Rather than being engaged with the direct mappingThe equation of the residual block can be formulated as follows:

$\phi output = \Phi act\ (\varphi Conv\ (i) + i)$

where $\varphi Conv(i)$ denotes the output of convolutional operation applied on the input of i and Φact denotes the activation function. In this work, we proposed three customized CNNS based on multiple residual blocks

1) Proposed 3-RBNet: 3-RBNet is comprised of three residual blocks having a total of 78 layers with a total number of 89 connections and 11.9 million parameters. Each residual block contains four parallel sets of layers connected at the end with

TABLE II
DESCRIPTION OF THE PROPOSED RESIDUAL BLOCK-BASED SELF-ATTENTION NETWORKS

Proposed model	No of layers	No of connections	Total parameters
3-RBNet	78	89	11.9 million
5-RBNet	123	142	12.3 million
9-RBNet	214	249	23.8 million

TABLE IV
INFORMATION OF ALL RESIDUAL BLOCKS OF THE PROPOSED 5-RBNET

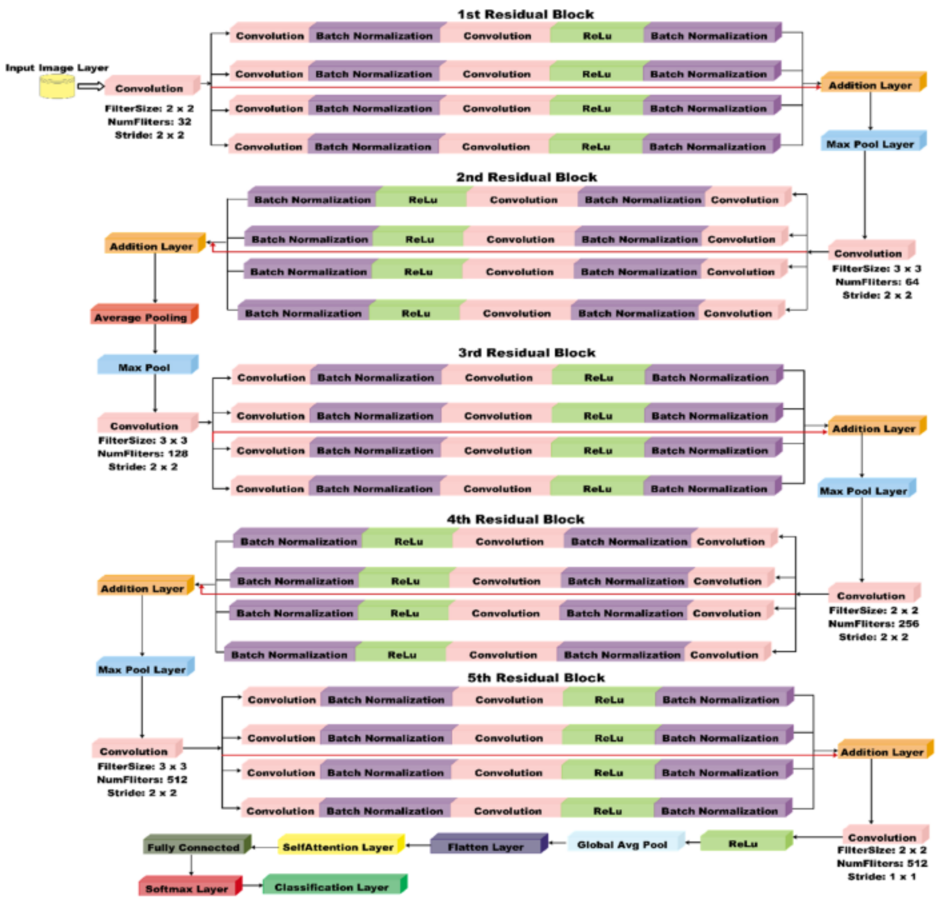
Residual block	Number of filters	Filter size
1st	64	2×2
2nd	128	3×3
3rd	256	3×3
4th	512	2×2
5th	1024	3×3

TABLE III
DESCRIPTION OF EACH RESIDUAL BLOCK OF DEPTH AND FILTER SIZE OF THE PROPOSED 3-RBNET

Residual block	Number of filters	Filter size
1st	64	3×3
2nd	256	3×3
3rd	512	3×3

an additional layer. The input layer of the model accepts an input of size 224×224×3 which is followed by a convolution layer containing 64 numbers of filters with size 3×3 and a step size of 2×2. Then, a residual block containing four parallel sets of layers is added, in which the first convolutional layer has 64 filters of size 2×2 and a stride of 1×1. Following that, a batch normalization layer is attached to improve the convergence of the proposed model as well as enhance the stability of the model during training. Another convolution layer containing 64 filters with a size of 2×2 and a stride of 1×1 is added with rectified linear unit (RELU) activation, which acts as an activation function to add nonlinearity in the proposed model. Moreover, another batch normalization layer is added in the same way as all four parallel sets of layers. Moreover, the other three residual blocks follow the same strategy as the first residual block, with different values of depth and filter sizes, which are illustrated in Table III. The last residual block is a convolution layer containing 512 filters with a filter size of 2×2 and a step size or stride of 1×1 with RELU layer. In addition, global average pooling is inserted, and a flattened layer is added to convert the multidimensional feature vector map obtained after pooling into an array of one dimension to implement the self-attention layer. In the end, a fully connected Softmax and classification layer is added to classify the disease. The created model was trained on selected datasets, and self-attention activation was utilized to extract the deep featuresThe sizes of extracted features were N × 512. The architecture of 3-RBNet is shown in Fig. 6. 2) Proposed 5-RBNet: The proposed 5-RBNet contains five residual blocks with a total of 123 numbers of layers with 142 total connections and 12.3 million parameters. Each residual block contains four parallel sets of layers connected to the addition layer. The network starts with the input layer, which takes a 224 × 224 × 3 size image. The first convolution layer contains 64 numbers of filters with a filter size of 2×2 and a stride of 2×2. The first residual block is added, which contains 4 parallel sets of layers starting from a convolution layer containing 64 total numbers of filters of filter size 2×2 and a stride of 1×1. After that, another convolution layer containing 64 filters of size 2×2 and a step size of 1×1 is used, which is followed by a RELU andbatch normalization layers. All other remaining parallel layers follow the same phenomena as followed by the first one. After that, the max pool layer is attached, which is used to obtain a feature map of maximum values; in this way, the dimensionality of the available data is reduced. Similarly, the remaining four residual blocks follow the same strategy as followed by the first residual block but with different depths, stride and filter sizes

The last residual block contains a convolution layer of 1024 filters with a filter size of 2×2 and stride of 1×1. After that, RELU activation is added. In the last, a GAP, flatten, selfattention, FC, Softmax, and classification layers have been added in order to complete the network. The 5-RBNet was trained on selected datasets, and prominent features are extracted from the self-attention activation. The size of extracted features was N × 1024. The architecture is visually presented in Fig. 7. 3) Proposed 9-RBNet: The proposed 9-RBNet contains nine residual blocks, having a total of 214 layers, 249 connections, and 23.8 million parameters. Each residual block contains four parallel structure layers connected to the addition layer. A selfattention layer is added after a series of residual blocks. The proposed model input size is 224 × 224 × 3. The first convolution layer applied with filter size 2×2, stride of 2×2, and with 32 number of filters. Then comes the first residual block which contains 4 sets of parallel layers, the first convolution layer of residual block is of filter size 2×2, stride of 1×1, and contains 32 numbers of filters. The second layer used in residual block is the batch normalization layer, and the third layer is again the convolution layer in the residual block containing 32 number of filters of size 2×2 and a stride of 1×1, then later on RELU and BN are used at the end of the residual block. RELU layer is added to the residual block which will act like an activation functionto introduce nonlinearity in the model; in this way, the proposed model will be able to learn complex features and patterns from the data more efficiently. BN is added to the residual block to improve the models stability, increase the convergence speed, and enhance the performance of the proposed model during the process of training. The other three parallel sets of layers which create the residual block have the same parameters as the first set of layers as discussed above. To connect the four parallel sets of residual block layers, an addition layer has been added. After that, a max pooling layer is added with pool size 5×5 and a stride of 1×1. The purpose of max pooling layer is to reduce the dimension of the input data by convolving the filter or kernel of window size according to the pool size of the input feature vector map and selecting the maximum value from every window. In this way, only the maximum value feature vector



map is obtained; hence, the dimensionality is reduced. The next convolution layer added contains 64 numbers of filters of size 3×3 and a stride of 2×2. All eight other residual blocks use the same mechanism and number of layers as the first residual block with different depth sizes. The depth and sizes of each residual block are illustrated in Table V. After the last residual block, a convolution layer containing 1280 number of filters with a filter size of 2×2 and a stride of 1×1 has been selected. After that, a RELU layer is added. A global average pooling layer is attached to condense all obtained feature vector maps into one feature map containing the average values from all other feature vectors. Moreover, a flattened layer is connected to convert the 2-D feature map into 1-D and pass this feature map to the self-attention layer. Following that, FC layer, Softmax, and a classification layers have been added to complete the network. The created network was trained on bothselected datasets and the features are extracted from the self-attention activation. The dimensions of extracted features are N × 1280

TABLE V
DESCRIPTION OF BLOCK WISE DEPTH IN THE PROPOSED 9-RBNET

Residual block	Number of filters	Filter size
1st	32	2×2
2nd	64	3×3
3rd	64	3×3
4th	128	2×2
5th	256	3×3
6th	512	2×2
7th	512	2×2
8th	1024	3×3
9th	1280	2×2

C. Proposed Models Learning and Features Extraction

The training process of the proposed DL models has been added under this section. In the training process, the enhanced images dataset has been divided into a ratio 50, 50. This means that the total 50% of the images have been employed for training and remaining used for the testing and validation. Several hyperparameters have been employed for the training of the models such as learning rate value of 0.00023, momentum value of 0.722, epochs are 50, mini-batch size of 64, and stochastic gradient descent (SGD) is employed as an optimizer. After the training, the obtained models have been employed for the features extraction. The features are extracted from the selfattention layers of all three models and performed classification using neural networks. The obtained results are compared with each other and the best accuracy model is selected for the further process such as optimization. In the optimization process, a binary dragon fly optimization algorithm has been implemented and applied on two feature vectors: 1) best model accuracy of wheat crop, and 2) best model accuracy of cotton crop

D. Binary Dragonfly Optimization for Features Selection

Feature selection refers to the process of selecting the most suitable, relevant, and informative maps from a given dataset inorder to achieve better classification results. Selection of features is a careful task because noisy, redundant, or useless features increase the complexity of the model, which increases the misclassification. In this research work, a swarm intelligence-based optimization is used, called dragonfly algorithm. The motion behavior of dragonflies inspires this algorithm. Two main stages of this optimization technique are exploration of target, i.e., food or prey, and exploitation. The two main behaviors of a swarm are attraction toward food and running away from the enemy, which depend on given factors for the positioning of an individual in a swarm. Mathematically, the behavior of the swarm is computed by using the following equations. The separation between two individuals is given by the following:

$$Ei = - Z\ b = 1\ Y - Yb \quad - (7)$$

where Y denotes the location of the current individual, Yb denotes the position of the bth neighbor, and Z is the number of dragonflies. Alignment of the individuals in a swarm is given by the following:

$$Ci = Z\ b=1\ Gb\ Z \quad - (8)$$

where Gb denotes the bth neighboring individual. The cohesion between individuals of a swarm is given by the following:

$$Ob = Z\ b=1\ Yb\ Z - Y \quad - (9)$$

To calculate the movement toward food, the target is formulated as follows:

$$Vi = Y + - Y \quad - (10)$$

where Y denotes the current position of the individual and Y + presented the position of food. Diversion from the enemy of any individual is mathematically defined as follows:

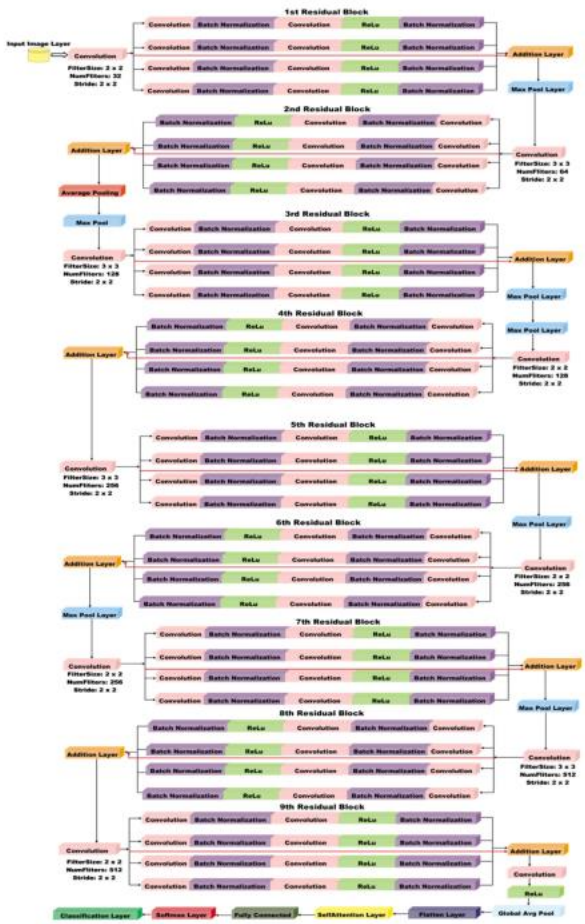
$$Di = Y - Y \quad - (11)$$

where Y the position of the individual is fly and Y − represents the position of the enemy. These above five parameters are considered the behavior-building factors of dragonfly behavior. There is also a need to improve the performance parameters of a dragonfly, like randomness, stochastic behaviors, and exploration capabilities, because the individuals in a swarm follow a random walk (Levy flight) behavior so there is no particular solution for neighboring fliers. To tackle this randomness problem, the position of dragonflies is updated using the following:

$$Yt+1 = Yt + \text{Lévy}\ (e) + Yt - \quad (12)$$

where t denotes the current iteration and e give the dimensions of position vector. The value of Levy flight can be calculated as follows:

$$\text{Lévy}(q)=0.01 \times u1 + \sigma\ |u2|\ \beta \quad - (13)$$



where u1 and u2 are two random numbers lie between the range of [0, 1], β is a constant value, and σ is mathematically formulated as follows:

$$\sigma = \left(\zeta\,(1+\beta) \times \sin\ \pi\beta\ 2\ \ \zeta\ 1+\beta\ 2\ \ \beta \times 2(\beta-1\ 2) \right)^{1/\beta} \quad (14)$$

where ζ (q)= (q − 1) and the fitness and cost value of dragonfly optimization is measured by employing the KNN classifier.

The KNN classifiers return fitness which is described in (15) and the cost value is returned using

(14)

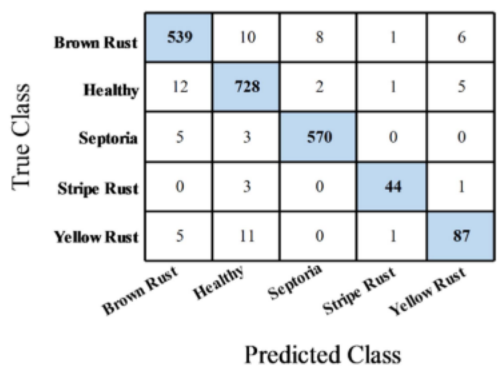
$$\mathcal{K}_{\text{fit}} = \left| \frac{FV\left(\mathfrak{y}_{\text{best}}\right)}{FV\left(\mathfrak{y}_i^k\right)} \right| \quad (15)$$

$$\mathcal{K}_{\text{cost}} = \omega_\alpha \times \sigma_{\text{err}} + \omega_\beta \times \left(\frac{\text{Num\_ of\_sel\_features}}{\text{Max\_}feat} \right) \quad (16)$$

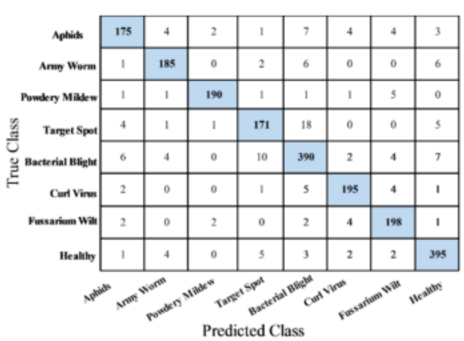
where ωα and ωβ presented the coefficient value which is 0.92 and 0.014, respectively. σerr The error rate is presented, whichis calculated using the following:

σerr = 1 − ψaccur.

best feature selection. The proposed feature selection algorithm is first applied on the 9-RBNET self-model for the wheat dataset and returns a feature vector of dimension N × 916. Also, the optimization algorithm is applied on 5-RBNET Self for cotton dataset and returns a feature vector of N × 736. The main reason behind the selection of both models is initial accuracy. These models give better accuracy on the selected datasets; therefore, we applied optimization. The selected features are passed to neural network classifiers such as narrow neural network (NN), bilayered NN, trilayered NN, medium NN, and wide NN. Furthermore, the hyperparameters of these neural networks are optimized using Bayesian Optimization. The results are discussed in Section



ig. 9. Confusion matrix of 3-RBNET Self for wheat dataset using wide NN.



CHAPTER 4

RESULT AND DISCUSSION

The results of the proposed methodology have been presented in this section. The selected datasets are divided into 50:50 ratios. The 50% samples are utilized for training and the remaining 50% data are used for testing purposes. The entire experimental process was carried out using 10-fold cross validation. For the training of proposed models, several hyperparameters are manually selected such as such as learning rate value of 0.00023, momentum value of 0.722, epochs are 50, mini-batch size of 64, and SGD is employed as an optimizer. The performance of the proposed models has been is conducted using several neural network classifiers such as narrow neural network (NN), bilayered NN, trilayered NN, medium NN, and wide NN. In addition, the hyperparameters of the models has been optimized using Bayesian optimization (BO). The hyperparameters for BO optimization is described below. The performance of each classifier is evaluated using accuracy, precision, sensitivity, FNR, time, Kappa, and Mathew's correlation coefficient (MCC) measures. The entire experiments are conducted using MATLAB R2023a on Desktop computer configured with 128GB RAM, 512 SSD, and 12GB NVIDIA RTX 3060 graphics card.

A. Results of the Proposed 3-RBNet Self The classification results of the proposed 3-RBNet on wheat dataset have been presented in Table VII. The proposed 3- RBNet) Self is implemented on wheat and cotton datasets to validate its performance. The table shows the maximum accuracy of96.40% on wide NN. Furthermore, few other measures are also computed such as sensitivity rate of 93.38, precision rate of 94.46, kappa value of 0.8868, and MCC value of 0.9292, respectively. Computational time is also computed and the minimum noted time is 10.196 (s) on medium NN classifier. Confusion matrix of the wheat dataset is also shown in Fig. 9. In this figure, the number of observations has been reported that can be utilized to confirm to above computed measures of wide NN classifier. Table VIII illustrates the results of cotton data using 3-RBNet Self architecture.

The highest accuracy of this dataset is 92.50% on wide NN classifier. In addition, the sensitivity rate of this classifier is 92.13, precision rate of 94.60, Kappa values of 0.6591, and MCC value of 0.9128, respectivelythe number of true and false observations has been added that can be utilized to confirm the calculated measures of wide NN classifier. The minimum noted computational time of this dataset for 3-RBNET Self architecture is 6.679 (sec) on medium NN classifier

B. Results of the Proposed 5-RBNet Self Table IX shows the results of 5-RBNet Self architecture on the wheat dataset. The table shows that the narrow NN achieved

TABLE VII
RESULTS OF THE PROPOSED 3-RBNET ON WHEAT DATASET

Method	Sensitivity (%)	Precision (%)	FNR (%)	Accuracy (%)	Time (s)	Kappa	MCC
Narrow NN	92.92	94.20	7.08	95.80	13.632	0.8699	0.9239
Medium NN	92.58	92.84	7.42	95.70	10.196	0.8669	0.9155
Wide NN	**93.38**	**94.46**	**6.62**	**96.40**	**12.419**	**0.8868**	**0.9292**
Bi-layered NN	92.42	92.38	7.58	95.80	15.309	0.8699	0.9128
Tri-layered NN	93.08	92.58	6.92	95.80	27.711	0.8684	0.9165

The bold values denote the best results.

TABLE VIII
RESULTS OF THE PROPOSED 3-RBNET ON COTTON DATASET

Method	Sensitivity (%)	Precision (%)	FNR (%)	Accuracy (%)	Time (s)	Kappa	MCC
Narrow NN	89.60	89.71	10.40	90.10	11.878	0.5455	0.8823
Medium NN	91.31	91.56	8.68	91.60	6.679	0.6168	0.9021
Wide NN	**92.13**	**94.60**	**7.86**	**92.50**	**7.739**	**0.6591**	**0.9128**
Bi-layered NN	90.40	90.61	9.60	90.80	8.953	0.5789	0.8917
Tri-layered NN	89.80	89.912	10.20	90.10	18.115	0.5478	0.8842

The bold values denote the best results.

TABLE IX
RESULTS OF THE PROPOSED 5-RBNET SELF ON WHEAT DATASET

Method	Sensitivity (%)	Precision (%)	FNR (%)	Accuracy (%)	Time (s)	Kappa	MCC
Narrow NN	90.88	91.44	9.12	95.90	9.7245	0.8714	0.9006
Medium NN	93.00	92.92	7.00	96.30	6.2061	0.8837	0.9198
Wide NN	93.46	93.68	6.54	**96.70**	6.4544	0.8975	0.9269
Bi-layered NN	94.34	94.04	5.66	96.70	9.7736	0.8959	0.9329
Tri-layered NN	93.00	92.52	7.00	96.20	20.116	0.8806	0.9174

The bold value denote the best results.

TABLE X
RESULTS OF 5-RBNET SELF ON COTTON DATASET

Method	Sensitivity (%)	Precision (%)	FNR (%)	Accuracy (%)	Time (s)	Kappa	MCC
Narrow NN	91.65	91.72	8.35	92.20	10.104	0.6436	0.9056
Medium NN	92.70	92.86	7.30	**93.20**	6.037	0.6903	0.9181
Wide NN	92.65	92.91	7.35	93.20	6.157	0.6924	0.9195
Bi-layered NN	90.98	91.05	9.02	91.50	8.265	0.6101	0.8978
Tri-layered NN	89.98	90.20	10.02	90.90	9.629	0.5834	0.8877

The bold value denote the best results.

an accuracy of 95.90%, the medium NN obtained an accuracy of 96.30%, the wide NN obtained an accuracy of 96.70%, 96.70% accuracy obtained by bilayered NN, and 96.20%

accuracy is achieved by TNN classifier, respectively. Based on these accuracies, it is noted that the wide NN classifier obtained the highest accuracy. Moreover, the other computed measures of this classifier are sensitivity rate of 93.46, precision rate of 93.68, Kappa value of 0.8975, and MCC value of 0.9269, respectively. A confusion matrix is illustrated in Fig. 11 that can be utilized to confirm these measures. The execution time of each classifier is also noted, and the minimum time of 6.2061 (s) for medium NN, whereas the wide NN is executed in 6.4544 (s). Table X presents the results of the proposed 5-RBNet Self for cotton dataset. This table demonstrates that the medium NN classifier achieved the highest accuracy of 93.20%. The sensitivity rate of this classifier is 92.70, precision rate of 92.86, Kappa value of 0.6903, and MCC value of 0.9181, respectively. The rest of the classifications also obtained better accuracy of 92.20%, 93.20%, 91.50%, and 90.90%, respectively. The confusion matrix is also illustrated in Fig. 12 for medium NN that can be utilized to confirm the reported measures. Time is also noted for the testing process of each classifier, and it is observed that the minimum reported time of 6.03 (s) for the medium NN classifier

D. Analysis of Model Accuracy The results of the three proposed models are 3-RBNET Self, 5-RBNET Self, and 9-RBNET Self. The results are presented in Tables VI–XI for wheat and cotton datasets. Moreover, the confusion matrices are illustrated in Figs. 9–14. Based on these results, we concluded that the 3-RBNET Self-model accuracy is not enough to match the accuracy of the 5-RBNET Self and 9-RBNET Self. The performance of the wheat dataset is better on 9-RBNET Self architecture, whereas the performance of cotton dataset is better on 5-RBNET Self architecture. Therefore, we selected 5-RBNET Self architecture and 9-RBNET Self architecture features and optimized using a binary dragonfly optimization. The purpose of optimization is to maintain the accuracy, precision, Kappa, and MCC value, whereas reduce the computational time

E. Optimization Results As seen in the analysis section, the 5-RBNet Self obtained the better accuracy on the wheat dataset (highest classification accuracy of 98.40% with an execution time of 5.6170 s on the medium NN classifier, see Table X). After employing the optimization algorithm on this model, the wide NN obtained an accuracy of 98.60% Table XII). The precision rate of this classifier is 98.44%, the sensitivity rate of 98.42%, the kappa value of 0.955, and MCC value of 0.980, respectivelyThe confusion matrix is also shown in Fig. 15. Compared to accuracy with original proposed network, the cotton dataset accuracy has been improved, and time is almost 100% reduced. The minimum noted time of 2.9939 (s) for wide NN classifier, whereas the computation time of medium NN is 3.4632 (s) before optimization (best time). Tables XIII and XIV presents the classification results of the optimization algorithm for cotton dataset on 5-RBNet Self architecture. As discussed under the analysis section, the 5-RBNet Self architecture performed well for the cotton dataset. Before optimization, the best accuracy and time of this network was93.205% and 6.037 (s). After employing the optimization algorithm, the obtained accuracy is 93.90% for wide NN classifier. Moreover, the computational time of this classifier is 3.068 (s). Overall, it is observed that the accuracy is improved after the optimization algorithm, and time is 100% reduced compared to the originally proposed architectures (before optimization). Fig. 16 illustrates the confusion matrix of a wide NN classifier that can be utilized to confirm the computed measures such as sensitivity, precision, Kappa, and MCC. 1) Proposed Results on EuroSAT Dataset: The proposed classification results using EuroSAT are presented in this section. Results are given in Table XV. In this table, it is shown that the trilayered NN obtained the highest accuracy of 83.10%. The sensitivity rate of this classifier is 81.35%, the precision rate is 81.22%, the Kappa value is 0.0593, and the MCC value is 0.7931, respectively. The confusion matrix of this classifier is illustrated in Fig. 17, which can be utilized to confirm the obtained performance measures. In addition, the time of each classifier has been noted, and the narrow NN classifier was executed in 188.57 (sec), which is faster than other classifiers’

F. Discussion and Comparison With SOTA

A brief discussion of the proposed method has been conducted under this section. In this article, we proposed deep residual self-attention models for the classification of wheat and cotton left diseases. We proposed three variants of residual blocks with a self-attention layer in order to extract the most prominent information from the data. The first model consists of three residual blocks, the second model consists of five residual blocks, and the third model contains nine residual blocks. All models have been trained through the selected datasets as discussed Fig. 17. Confusion matrix of the proposed method for EuroSAT dataset.

Validation accuracy of the proposed model on each epoch. previously (see Section II-A1), and the validation accuracy at each epoch was measured, as shown in Fig. 18. The proposed model was trained using 50 epochs. The table illustrates that the 3-RBNet Self has lower validation accuracy than the 5 and 9-RBNet Self. The accuracy of 5-RBNet Self was stable after 36 epochs. Initially, the loss was high for the proposed 9-RBNet, but the validation accuracy improved as the epoch passed. After the experimental procedure, it was observed that the rest of the network outperformed the proposed 9-RBNet. In addition, a detailed comparison is conducted with pretrained neural networks, as shown in Fig. 19. It represents the comparison of the proposed models for both cotton and wheat datasets with other state-of-the-art ML models. Models involve VGG-19, AlexNet, ResNet-18, ResNet-50, ResNet-101, and NasNet-Mobile, and they are compared with the proposed models, namely 3-RBNET Self, 5-RBNET Self, 9-RBNET Self and optimized 9-RBNET Self models. It is shown that the proposed architectures obtained better classification accuracy. Moreover, Fig. 20 illustrates the proposed labelled results. Finally, a comprehensive comparison is conducted with the state-of-the-art techniques, as presented in Table XVI. The table describes that the 2023 study achieved the highest accuracy of 98.5% with the method of continuous learning for wheat disease. At the same time, our proposed framework achieved 98.64% accuracy on

TABLE XVI
COMPARISON WITH THE SOTA TECHNIQUE

Refs.	Year	Dataset	Methodology	Accuracy
Alharbi et al. [69]	2023	Self-created dataset	Classification of wheat disease using continual learning	93.19%, 98.5%
Long et al. [70]	2023	Wheat growth greenhouse dataset	Classification of wheat disease using DL methods	97.05%
Nigam et al. [71]	2023	Wheat rust disease dataset	Wheat disease identification using Deep transfer learning method	97.8%
Dhakal et al. [72]	2023	Hyperspectral of wheat disease	Damaged wheat analysis using ML method	97.00%
Jenifa et al. [73]	2019	Cotton leaves disease	Classification of cotton leave detection using DCNN	96%
Alexnet Model (TL)	2024	EuroSAT	TL based training and features extraction	80.52
Resnet101 Model (TL)	2024	EuroSAT	TL based training and features extraction	81.46
Inception V3 Model (TL)	2024	EuroSAT	TL based training and features extraction	82.95
Proposed methodology (Wheat and Cotton)				**98.60%, 93.90%**
Proposed methodology (EuroSAT Dataset)				**83.10%**

CHAPTER 6

CONCLUSION

In this article, we proposed a novel self-attention and optimization architecture for crop leaf disease classification. The strength of the proposed architecture is designing two selfattention 5-RBNET and 9-RBNET architectures for cotton and Fig. 20. Visual illustration of proposed labelled images. wheat disease recognition. The contrast enhancement technique has been proposed based on the fusion of two filter mathematical formulations and passed resultant images to proposed architectures for the training. Based on the initial accuracy, the binary dragonfly optimization algorithm was applied on 5-RBNET and 9-RBNET architectures, and the best-selected features for the classification were obtained. Furthermore, hyperparameters of the neural network classifiers have been optimized using the Bayesian optimization algorithm. The proposed architecture obtained improved accuracy of 98.60 and 93.90% for wheat and cotton leaf diseases, respectively. Based on the detailed experimental process, we concluded the following.

1) The addition of a self-attention layer in 5-RBNET improved the accuracy and precision rate for cotton leaf disease recognition.

2) The addition of a self-attention layer in 9-RBNET improved the accuracy and precision rate for wheat leaf disease recognition

3) Optimization of extracted deep features and hyperparameters improved the accuracy and precision rates while reducing the computation time.

The limitation of the proposed framework was the manual setting of hyperparameters. In the future, we will propose a technique to dynamically select the hyperparameters, and an inverted bottleneck architecture with a self-attention layer will be proposed for the recognition of fruit leaf disease recognition

REFERENCES

[1] S. K. Upadhyay and A. Kumar, “A novel approach for rice plant diseases classification with deep convolutional neural network,” Int. J. Inf. Technol., vol. 14, pp. 185–199, 2022.

[2] X. Chao, “Toward sustainability: Trade-off between data quality and quantity in crop pest recognition,” Front. Plant Sci., vol. 12, 2021, Art. no. 811241.

[3] A. Arshaghi, M. Ashourian, and L. Ghabeli, “Potato diseases detection and classification using deep learning methods,” Multimedia Tools Appl., vol. 82, pp. 5725–5742, 2023.

[4] S. Chakraborty and A. C. Newton, “Climate change, plant diseases and food security: An overview,” Plant Pathol., vol. 60, pp. 2–14, 2011.

[5] J. Nie, Y. Wang, and X. Chao, “Artificial intelligence and digital twins in sustainable agriculture and forestry: A survey,” Turkish J. Agriculture Forestry, vol. 46, pp. 642–661, 2022.

[6] J. B. Ristaino et al., “The persistent threat of emerging plant disease pandemics to global food security,” Proc. Nat. Acad. Sci., vol. 118, 2021, Art. no. e202223911

[7] E.-C. Oerke, “Crop losses to pests,” J. Agricultural Sci., vol. 144, pp. 31–43, 2006.

[8] C. Yang, “Remote sensing and precision agriculture technologies for crop disease detection and management with a practical application example,” Engineering, vol. 6, pp. 528–532, 2020.

[9] S. J. Janssen et al., “Towards a new generation of agricultural system data, models and knowledge products: Information and communication technology,” Agricultural Syst., vol. 155, pp. 200–212, 2017.

[10] A. Buchelt et al., “Exploring artificial intelligence for applications of drones in forest ecology and management,” Forest Ecol. Manage., vol. 551, 2024, Art. no. 121530.

[11] N. TeBlunthuis, V. Hase, and C.-H. Chan, “Misclassification in automated content analysis causes bias in regression. Can we fix it? Yes we can!,” Commun. Methods Measures, vol. 11, pp. 1–22, 2024.

[12] J. R. Lamichhane, M. P. You, V. Laudinot, M. J. Barbetti, and J.-N. Aubertot, “Revisiting sustainability of fungicide seed treatments for field crops,” Plant Dis., vol. 104, pp. 610–623, 2020. [13] S. Chohan, R. Perveen, M. Abid, M. N. Tahir, and M. Sajid, “Cotton diseases and their management,” in Cotton Production and Uses: Agronomy, Crop Protection, and Postharvest Technologies. Berlin, Germany: Springer, 2020, pp. 239–270.

[14] W. B. Demilie, “Plant disease detection and classification techniques: A comparative study of the performances,” J. Big Data, vol. 11, 2024, Art. no. 5.

[15] J. J. Casanova, S. A. O’Shaughnessy, S. R. Evett, and C. M. Rush, “Development of a wireless computer vision instrument to detect biotic stress in wheat,” Sensors, vol. 14, pp. 17753–17769, 2014.

[16] R. Manavalan, “Automatic identification of diseases in grains crops through computational approaches: A review,” Comput. Electron. Agriculture, vol. 178, 2020, Art. no. 105802.

[17] M. Alessandrini, R. C. F. Rivera, L. Falaschetti, D. Pau, V. Tomaselli, and C. Turchetti, “A grapevine leaves dataset for early detection and classification of esca disease in vineyards through machine learning,” Data Brief, vol. 35, 2021, Art. no. 106809.

[18] M. O. Ojo and A. Zahid, “Improving deep learning classifiers performance via preprocessing and class imbalance approaches in a plant disease detection pipeline,” Agronomy, vol. 13, 2023, Art. no. 887.

[19] A. V. Panchal, S. C. Patel, K. Bagyalakshmi, P. Kumar, I. R. Khan, and M. Soni, “Image-based plant diseases detection using deep learning,” Mater. Today: Proc., vol. 80, pp. 3500–3506, 2023.

[20] D. Jones, “Disease and pest constraints to banana production,” in Proc. III Int. Symp. Banana: ISHS-ProMusa Symp. Recent Adv. Banana Crop Protection Sustain. 828, 2007, pp. 21–36.

[21] Ü. Atila, M. Uçar, K. Akyol, and E. Uçar, “Plant leaf disease classification using EfficientNet deep learning model,” Ecological Informat., vol. 61, 2021, Art. no. 101182.

[22] H. Firat, M. E. Asker, M. ˙I. Bayindir, and D. Hanbay, “3D residual spatial–spectral convolution network for hyperspectral remote sensing image classification,” Neural Comput. Appl., vol. 35, pp. 4479–4497, 2023.

[23] Z. Lv, P. Zhang, W. Sun, J. A. Benediktsson, J. Li, and W. Wang, “Novel adaptive region spectral-spatial features for land cover classification with high spatial resolution remotely sensed imagery,” IEEE Trans. Geosci. Remote Sens., vol. 61, 2023, Art. no. 5609412.

[24] V. Vapnik, “Support-vector networks,” Mach. Learn., vol. 20, pp. 273–297, 1995.

[25] E. Fix and J. L. Hodges, “Discriminatory analysis. Nonparametric discrimination: Consistency properties,” Int. Stat. Rev./Revue Internationale de Statistique, vol. 57, pp. 238–247, 1989.

[26] J. Redmon, Darknet: Open Source Neural Networks in C, 2013. [Online]. Available: http://pjreddie.com/darknet/

[27] M. Tan and Q. Le, “Efficientnet: Rethinking model scaling for convolutional neural networks,” in Proc. Int. Conf. Mach. Learn., 2019, pp. 6105–6114.

[28] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “Imagenet classification with deep convolutional neural networks,” in Proc. Adv. Neural Inf. Process. Syst., 2012, vol. 25.

[29] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2016, pp. 770–778.

[30] X. Li, Q. Ding, and J.-Q. Sun, “Remaining useful life estimation in prognostics using deep convolution neural networks,” Rel. Eng. Syst. Saf., vol. 172, pp. 1–11, 2018. [31] M. A.-M. M. Salem, “Multiresolution image segmentation,” 2008