IIETA International Information and Engineering Technology Association
Advancing the World of Information and Engineering

Traitement du Signal
Vol. 40, No. 5, October, 2023, pp. 2051-2062
Journal homepage: http://iieta.org/journals/ts

A Capsule Attention Network for Plant Disease Classification

Ponugoti Kalpana*, R. Anandan

Department of Computer Science and Engineering, Vels Institute of Science, Technology and Advanced Studies, Chennai 600117, Tamil Nadu, India

Corresponding Author Email: kalpanaraogonait@gmail.com

https://doi.org/10.18280/ts.400523

Received: 27 March 2023
Revised: 26 August 2023
Accepted: 12 September 2023
Available online: 30 October 2023

Keywords:
artificial intelligence, machine learning, deep learning, plant diseases, capsule feedforward networks, attention layers

ABSTRACT

The identification of plant diseases is one of the most essential and difficult concerns in agriculture, necessitating solutions with a brighter light. With the onset of artificial intelligence (AI), machine learning (ML) and deep learning (DL) algorithms have aided farmers in identifying and classifying plant features with a high degree of intellectual precision. However, accurate disease classification in plants is essential for empowering farmers to cultivate more and produce more. This study therefore presents a unique assembly of attention, capsule, and feedforward classification layers for reaching the maximum classification accuracy for plant diseases. The proposed system uses user-defined customized Convolutional Transfer Learning networks (CTLN) to extract features and the attention networks exclude unnecessary features and highlight only critical features for classification. Finally, the selected characteristics are sent to the Feedforward Capsule networks to improve performance. This paper proposes a paradigm that overcomes the constraints of existing deep learning networks and drastically decreases the computing burden. The suggested network is thoroughly evaluated utilizing Plant Village databases containing over 50,000 photos of healthy and diseased plants. The performance metrics of the proposed method are evaluated and compared to those of other learning networks. Compared to previous models, experimental results indicate that the proposed model has a 99.8 percent accuracy rate, lending support to the new categorization method that benefits farmer well-being.

1. INTRODUCTION

Agriculture is the true foundation of economic progress in every nation. In the past, numerous agronomists and researchers were drawn to identify the various agricultural challenges due to the predicted exponential demand for food and the scarcity of basic agricultural resources. Plant diseases and crop losses are considered complexities that have an impact on yields and may subsequently lead to economic losses as well [1-5]. Recently, applications of the machine and deep learning have skyrocketed after successful implementation in agricultural applications such as crop detection, plant disease identification, crop plant monitoring, and management. Convolutional Neural Networks (CNN) have gained the momentum in agricultural domain specifically applied the plant disease classification. Additionally, pre-trained models such as Alexnet [6], GoogleNet [7], VGG- l9 [8], and Resnets-50 [9] are also tested on plant disease datasets to produce the best performance of detection. However, these existing models still lack robustness across a large number of datasets.

In recent years, Capsule networks [10] have gained more popularity than CNN in different applications such as plant disease detection, text classification, tumor classification, bioinformatics, and simple classification problems. In contrast to CNN, which encodes information in a scalar manner, capsule networks are groups of neurons that encode and store spatial information in vector form. Though the Capsule networks are used for various applications, these networks do not learn the important local features from multiple scale plant images that may increase the false rate of detection. To advance agriculture, particularly in the classification of plant diseases, a new design that will assist in overcoming the restrictions is urgently sought.

Motivated by this problem, this paper presents the significant improvisation in Capsule Networks to achieve the better classification of plant diseases. The paper's primary contribution is as follows:

Concern with the novelty of Capsule networks, this paper presents a novel architecture that embeds the attention layers between the convolutional and primary capsule layers. The attention networks solve the aforementioned problem by learning the local features while capsule networks learn from the spatial correlation between the features. Moreover, the inclusion of attention layers in Capsule networks has reduced the computational complexity when compared with other existing learning algorithms.

- This paper also presents the novel idea of replacing the conventional backpropagation training network with powerful feedforward layers that work based on the theory of Extreme Learning Machines (ELM). ELM is used as the final layer of classification with high learning speed and low false rate detection. Additionally, the usage of ELM also keeps computational efficiency while having a large receptive field of plant datasets.
- The paper presents the extensive experiments carried out to evaluate the proposed framework and

demonstrate that the proposed model was able to supply better performance and robust and stable classification across larger datasets. In general, the proposed framework has outperformed the other existing deep learning and capsule networks in the detection of plant diseases.

The remaining sections are organized as follows:

In Section 2, various papers on intelligent techniques for plant disease diagnostics are discussed briefly. In Section 3, it is detailed how the suggested model functions and how datasets are gathered. Sections 4, 5, and 6 present respectively, the implementation details, experiments, and outcomes. The article is finally ended in Section 7, along with suggestions for improvement.

2. RELATED WORKS

Azadbakht et al. [11] examined the display to determine how to recognize wheat leaf rust at high, medium, and low levels of leaf area index (LAI), as well as at the shadow scale. The exhibition was broken down using four different approaches, including Support Vector regression, Random Forest (RF) regression, vs. Gaussian process regression, and an improved regression tree. This also accounts for the illness's level of severity. The study used to analyze the presentation focused on 7000 hectares of wheat development in the northwestern part of Iran. It makes use of hyperspectral reflectance data collected under varied natural conditions. Hyperspectral images can be used to identify wheat rust based on the results of the experiment. According to a correlation between spectral vegetation indices (SVIs) and machine learning (ML), ML outperforms SVIs [11].

Park et al. [12] recommended a component selection method known as "minimal redundancy and maximum relevance" (mRMR). Since PCA reduces dimensionality without revealing the most important spectral band. This method enables the selection of a rough band within an odd image. Along with convolutional brain organization and an associated network, a deep brain network was also suggested. The apple leaf is classified using FCN based on its state, namely common, late, youthful, early, and malnutrition. This method lessens the issue that arises in the hyperspectral picture, resulting in a disease forecast. The framework reduces the 519 groups in the hyperspectral image to 5 using automated feature selection (mRMR), and the leaf state arrangement is done using a deep neural network (VGG net + FCN) [12].

The method suggested by Karadağ et al. [13] for dealing with peppering fusarium disease helps. Using a phantom radiometer, plant light reflection aids in obtaining information about the plant. Mycorrhizal parasite, fusarium ailing, sound, and mycorrhizal leaves are the four types of leaves used for it. Between 350 and 2500 nm is where the frequency should fall. Two degrees of handling are involved. Highlight of a vector, and a collection of vectors. Three techniques, KNN, and NB are used with the end objective of order. It is established that KNN achieves a typical success rate of 100 percent, compared to ANN's success rate of 97.5 percent and Nave Bayes' success rate of 90 percent. The execution of grouping is slowed down by a lot of information. Using Wavelet Transformation (WT), the aspect can be lessened. Inteligencia Artificial 63(2017) 9 employs Sym5, db2, and Haar for this purpose. In ANN, backpropagation calculations are set up with order in mind. Ten field cross-approvals are used to test its appearance. To do the arrangement computation, Mat lab R2015b is used. When K=2, the db2, Haar, and sym5 accomplished better results. The typical achievement rates are 92.5%, 91.5%, and 85% for K=5. KNN produces better outcomes in comparison to other calculations. The elements are reduced via wavelet degradation from 2150 to 75 [13].

Iqbal et al. [14] mainly focus on the many illnesses that affect citrus plants and how they are grouped. Additionally, it provides a comprehensive description of the many methods utilized for the division cycle, highlights extraction, and includes selection, picture processing, and order methodology. Additionally, it discusses the electronic tools used for identification and characterization. Sicknesses that affect citrus plants include ulcers, black spots, citrus scabs, melanosis, and equipping. The methods used for the various stages of the inquiry are compared, and the K-mean computation is used in the current study to extract the illnesses. The Gray Level Co-Occurrences Matrix (GLCM) and the Back Propagation Neural Network were utilized to perform computations on and arrange various highlights (BPNN). Preprocessing, variety-based change, picture upgrading, sound reduction, sound resizing, and division techniques are investigated. Various methods of element extraction depend on the surface, variety, and form. It provides an overview of several classifier strategies along with examples of how they are used. According to the analysis, the pre-handling technique enhances division accuracy [14].

Considering extensive learning, Barbedo devised a method for picture grouping. When a data set is not available, an information expansion technique helps the plant with imaging. This essay focuses mostly on identifying a single sore or area rather than considering the full leaf in search of recognized evidence. The accuracy is 12% higher when using simply pain and spot than when utilizing full leaf. The full explanation of the innovative design utilized for plant disease diagnosis, the location of the data collection for differentiating the plant infection, and its accuracy after ID are all referred to. It also rattles off in a clear manner the list of complexities present in the plant example. Various kinds of images were used in the analysis and they are, 1. Image without any alteration 2. A picture without basis 3. A larger dataset Exactness is established for both original and expanded images [15].

Tavakoli and Gebbers presented a camera-based evaluation of water in the field and nitrogen analysis of winter wheat. This evaluation was conducted over three years (2012, 2013, and 2014). In the field, nitrogen treatment and various water concentrations are used to conduct the analysis. To conduct research, two AI calculations—specifically, RF and Partial Least Square Regression were created. The outspread estimate was performed using a phantom radiometer. Additionally, the Vegetation Index (VI) is computed independently. The R2 (RMSE) model is used to separately calculate the nitrogen content for the two types of information. Consolidated data information works better using arbitrary woods computation. Utilizing the computerized camera improves the performance of the calculation for nitrogen evaluation. It may also be synced with a mobile device. Only three unearthly groups can be reached, hence the scope of the analysis of plant state is also constrained [16].

Zhang et al. [17] developed a new DL-based system for tomato crop disease classification. Scientists developed a custom-tailored Faster-RCNN methodology in which the profound remaining system was utilized to automatically extract rather than the VGG16 model. Furthermore, the edge pixels were grouped using the k-means clustering method. This approach produces improved disease diagnosis findings for tomato crops, albeit at the cost of increased financial impact [17].

Verma et al. [18] studied the capsule networks to identify the disinfected plants. The author adopted the standard capsule network which plays a crucial part in the agriculture field in terms of "texture, orientation, and pose" more accurately than deep learning algorithms. Capsule network architecture collects every feature into a similar capsule group to frame the entire network. Though the capsule network achieved better accuracy than the CNN model, the model is not validated with segmented or real-time datasets [18].

Peker [19] proposed the ensemble capsule network with five channels for plant disease detection. Gabor, PCA, filters were combined with the capsule network to achieve the best performance in detecting the texture, shape, or leaf-affected area more accurately. The limitation of this network is high training complexity and overfitting problems arise at the pre-processing level [19].

Altan [20] reduced capsules on leaf images, and the CapsNET architecture was given to test the algorithm's usefulness in detecting plant leaf diseases. Plant leaf diseases are widespread, wreaking havoc on agricultural harvests and yields. Even tiny stains can alter seed dressing time and length, and Caps NET can do in-depth analysis on them.

Table 1. Quick summary of the literature survey

Authors	Proposed Method	The dataset used and Experimental Setup details	Results Achieved and Advantages	Demerits
Azadbakht et al. [11]	Boosted Regression Trees (BRT), RF Regression (RFR), Gaussian Process Regression (GPR), and Support Vector Regression (v-SVR)	Wheat leaf rust disease inversion based on canopy hyperspectral data were utilized. The Moghan fertilized plain, in the northwestern part of Iran, is where field tests were carried out. This plain typically sees the yearly cultivation of wheat over an area of around 7000 hectares.	Exhibited improved LAI = 0.91 for high, medium, and low and RMSE (8.5%) findings.	Since the training and test sets were derived from the same dataset, the results might not have been as impressive if the sources were different.
Park et al. [12]	minimum redundancy and maximum relevance (mRMR) for feature selection and CNN for classification	The indoor hyper-spectral dataset is used. The Python Theano library on the 'GeForce GTX TITAN X' Graphic Processing Unit for experimentation.	Excellent classification for both RGB and hyper-spectral images and also reduced the complexity.	Special attention is required in this framework for memory requirements
Karadağ et al. [13]	KNN is used for the classification	The pepper plants (Capsicum annuum) utilized in this study were cultivated in a climatic chamber at the plant protection lab of the Sanlurfa provincial GAP Agricultural Research Institute (GAPTEAM), Sanlurfa, Turkey.	With 91% classification accuracy, the KNN approach produced the best estimate results.	The trials only used clipped leaves; they did not use pictures of leaves in real settings.
Iqbal et al. [14]	discussed difficulties in identifying and classifying citrus plants	The review covers the different feature extraction techniques under different plants and experimental scenarios.	Almost all methods and instruments for identifying and categorizing plant diseases were covered by this framework.	According to this assessment, it is still difficult to simultaneously identify plant problems and take pictures under every conceivable circumstance. The research does point out that it is unreasonable to anticipate that an autonomous illness detection system will operate with 100 percent accuracy in real-world settings.
Bhyrapuneni and Rajendran [15]	Deep Neural Network and Transfer Learning are utilized	The images in the database were captured using several different sensors (smartphones, compact cameras, DSLR cameras) then Matlab was utilized for the experimentation.	Accuracy is improved by 12% and no crop had accuracy below 75% even though the framework is tested under 10 diseases.	the database does not cover the entire range of practical possibilities
Kumar and Anandan [16]	PLSR and RF are utilized	Leibniz Institute for Agricultural Engineering and Bioeconomy e.V. (ATB), Potsdam, Germany, served as the site for this study. A HP Z840 Workstation running 64-bit Windows 10 and outfitted with two Intel® Xeon® CPU E5-2667 v4 (3.2 GHz, 25 MB cache, 8 cores, Intel® vProTM), 256 GB of RAM, and two graphic cards (NVIDIA Quadro P5000) was utilized to process the images.	Provided better results in terms of RMSE Nitrogen content = 0.25–0.14 Water content = 0.75–0.36 Single date data = 0.24–0.15 Combined date data = 1.49–1.66	The training and testing were conducted using the same dataset, and all photos included leaves that were facing up on a uniform backdrop.

Zhang et al. [17]	RCNN and VGG16 architecture are integrated	The training dataset is from AIChallenger (https://challenger. ai/competition/pdr2018), which is laboratory data. An NVIDIA GeForce GTX 1060 GPU with 12 GB of RAM has been used to test the Faster RCNN model, which is based on TensorFlow. It uses Ubuntu as its operating system.	Compared to the original Quicker RCNN, the crop leaf disease detection exhibited a faster detection speed and 2.71% greater identification accuracy.	The authors underlined that segmentation and background removal had a significant influence on the findings and should be included in the model despite the low accuracy for some classes.
Verma et al. [18]	Capsule network modules	PlantVillage dataset is used. Python 3.7 was used to carry out the implementation on a workstation with a GTX 1060 6GB GPU.	Achieved 91.83% accuracy when compared to the traditional CNN model.	This framework leads to low accuracy when it is tested under real scenarios.
Peker [19]	An ensemble of capsule networks has been developed.	The tests are carried out in Python on a high-performance computer with an NVIDIA GeForce RTX 2070 Graphics Processing Unit (GPU), and the data set utilized in this study comprises pictures of nine distinct infected and healthy tomato crops.	Better accuracy is achieved (92.5%).	The main limitation is that the data sets consist of a small number of images and if the data set size increases the results are not maintained as the same.
Altan [20]	Capsule Networks	A plant village dataset is used. An NVIDIA GeForce GTX 1060 GPU with 12 GB of RAM has been used	Accuracy = 95.76%, Sensitivity = 96.37%, and specificity = 97.49%	The plant images are tested under both shadow and sun shining. But if the heterogeneity in the plant image increases the performance is degraded.
Waweru et al. [21]	CapsNet-SVM	A plant village dataset is used. Python 3.7 was used to carry out the implementation on a workstation with a GTX 1060 6GB GPU.	Achieved classification accuracy of 93.41%	This framework increased the time complexity for large-scale datasets.

The suggested CapsNET model sought to evaluate the applicability of different feature learning techniques for bell pepper crops and to improve the learning capacity of DL models. The CapsNET was fed the healthy and sick leaf pictures [20].

Waweru et al. [21] adopted the capsule network for classifying plant diseases. A new model based on the merging of CapsNet and support vector machine (CapsNet-SVM) was studied to classify illnesses in tomato leaves. The SVM model was utilized as a robust classifier, while the capsule network was optimized for feature extraction. The major goal was to improve SVM classification by employing artificial features retrieved by the capsule network model. The CapsNet-SVM model was found to be capable of autonomously extracting and classifying features from raw images. When compared to CNN, capsule networks, and CapNet-SVM, both frameworks are better in terms of accuracy but have poor performance in terms of spatial capture relationships [21]. Table 1 presents the summary of the literature survey.

From the related works, the research gap is found that the existing frameworks lack accurate identification of plant disease and those frameworks have high computational complexity. This motivates us to develop an intelligent framework for plant disease classification.

3. PROPOSED METHODOLOGY

A highly effective hybrid model is suggested to forecast plant diseases from the supplied images. Figure 1 shows the hybrid model's entire process flow. The proposed hybrid framework combines three layers such as convolutional attention layers, capsule networks, and finally feedforward classification layers.

The lightweight bi-layered self-attention network is sandwiched between the convolutional layers and capsule layers. To improve the classification of plant diseases, the features collected from the attention layers are then sent to the capsule layers, followed by the potent ELM layers. End-to-end learning eliminates the requirement for pre-processing or involved feature extraction processes by substituting a single ensemble deep neural network for a pipeline of learning components. The following sub-section explains the datasets, a general overview of Capsule networks, Feedforward classification layers, and proposed architecture.

3.1 Materials and methods

To facilitate the development of mobile disease diagnoses, Plant Village, an open-access photo database on plant health, was used for training and testing in this study. The 54,306 images included in the Plant Village dataset depict 14 distinct plant species. There are 38 classes in all, of which 26 show diverse plant diseases and 12 show classes of different plants with healthy leaves. Table 2 Contains information about the entire dataset.

3.2 Data augmentation

Figure 2 presents the visual representation of healthy and diseased sample plants. The image augmentation approach is implemented in the suggested design for efficient training and resolving the overfitting and class imbalance issues. The use of data augmentation is the most effective and efficient solution to this issue. Each image goes through a sequence of changes throughout the data augmentation step, creating a vast number of newly corrected training image examples. An affine transformation is used for effective data augmentation, as mentioned in study of Pei and Hsiao [22]. Translation, scaling, and rotation are examples of affine transformation techniques. This step is advised to avoid overfitting issues because most training picture samples obtained through the Augmentation procedure correlate. The proposed algorithm is then fed the enhanced data for an accurate categorization of various plant diseases.

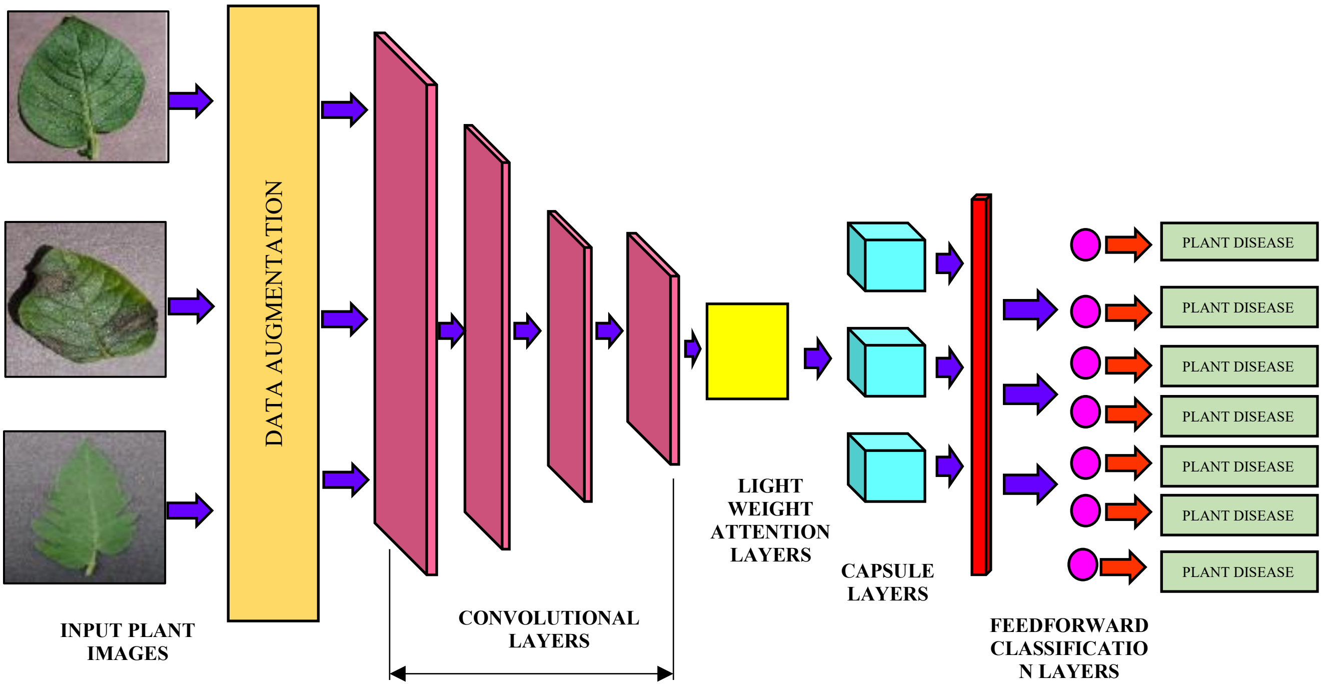


Figure 1. Proposed architecture for the multiple class plant diseases

Table 2. Complete descriptions of plant village datasets used in the proposed research

Plant Name	Types of the Plants	Class Label	No. of Samples
Tomato	Bacterial spot	1	28,226
	Early blight	2	
	Healthy	3	
	Late Blight	4	
	Leaf Mold	5	
	Septoria leaf spot	6	
	Spider Mites	7	
	Target spot	8	
	Mosaic virus	9	
	Yellow leaf curl virus	10	
Apple	Apple scab	11	3173
	Black rot	12	
	Cedar Apple rust	13	
Blueberry	Healthy	14	1502
Cherry	Healthy	15	7029
	Healthy	16	
	powdery mildew	17	
Corn	Gray leaf spot	18	4089
	Common rust	19	
	Healthy	20	
	Northern leaf blight	21	
Grape	black rot	22	12890
	Esca black measles	23	26782
	Healthy	24	
	Leaf blight	25	
Orange	Huanglongbing	26	1201
Peach	Bacterial spot	27	902
	Healthy	28	
Pepperell	Bacterial Spot	29	2503
	Healthy	30	
Potato	Early blight	31	12901
	Healthy	32	
	Late blight	33	
Raspberry	Healthy	34	1290
Soybeans	Healthy	35	890
Squash	Powdery Mildew	36	5690
	Healthy	37	
	Leaf scorch	38	

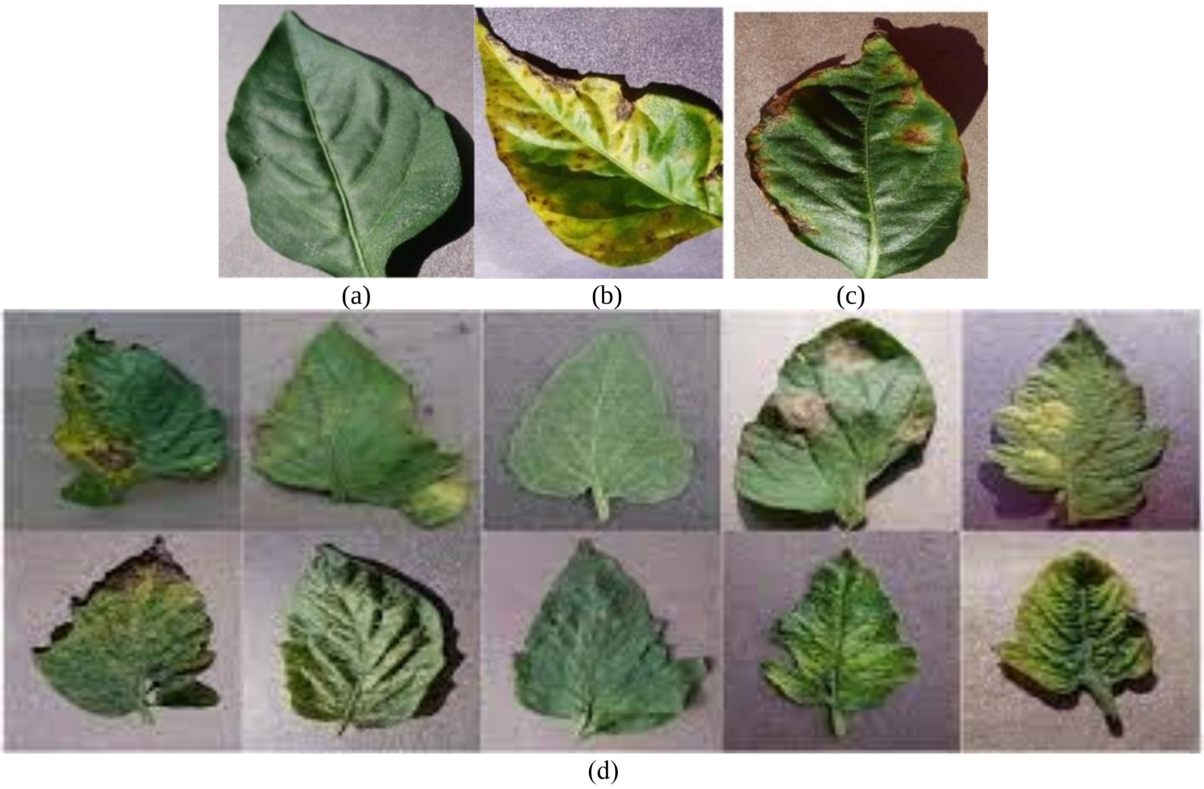


Figure 2. Visual representation of healthy and disease sample plant a) Healthy apple b) Pepper bell-bacterial spot c) Apple black rot d) Tomato diseases

The suggested architecture uses an image augmentation approach for efficient training and to address the overfitting and class imbalance issues. The use of data augmentation is the most effective and efficient solution to this issue. During the data augmentation step, each image is subjected to a variety of changes that generate a vast number of newly corrected training image examples. According to study of Pei and Hsiao [22], an affine transformation is utilized to effectively supplement data. Translation, scaling, and rotation affine transformation strategies are used. This step is recommended to avoid overfitting issues, as most training image samples acquired by the Augmentation method exhibit a correlation. The additional information is then loaded into the suggested algorithm for an accurate classification of various plant diseases.

3.3 Capsule network overview

Modern Capsule networks provide several advantages over traditional CNN. Convolutional layers, basic capsule layers, and classification layers make up the three layers of the capsule network. Squashing, where vector capsules are used in place of scalar outputs, and dynamic routing, which replaces max-pooling layers, are the two main characteristics of capsule networks that outperform conventional CNNs. Routing-by-agreement refers to the ability to pick and choose which parent in the layer above the capsule will receive the message. The capsule network can alter the connection strength for each optional parent. In squashing, capsule networks' output is compressed into a single ‘sector' rather than being routed through non-linearity individually as in CNNs. The capsule networks can capture spatial relationships. Without using any max-pooling layers, the squashing function improves the visualization of the likelihood that an object is present in the input image. The matrix of input vectors is multiplied by the weight matrix. The necessary spatial relationship between the image's low-level and high-level properties can be encoded.

$$Y(i.j) = W_{i,j}\, U(i,j) * S_j \tag{1}$$

To determine which capsule of a higher level the current capsule will forward its output to the weighted input vectors are summed.

$$S(j) = \sum_j Y(i,j) * D(j) \tag{2}$$

The squash function is then used to implement non-linearity. The squashing function preserves a vector's direction while mapping it to a minimum length between 0 and 1 and a maximum length of 1.

$$G(j) = squash(S(j)) \tag{3}$$

3.4 Feedforward layer

Traditional training networks are replaced in the proposed design by feedforward networks based on the ELM principle. Domathoti et al. [23] proposed the ELM classification of neural networks. This form of neural network employs a single hidden layer that does not require tuning. ELM performs better, is faster, and has less computational cost than other learning algorithms like SVM and RF. ELM makes use of the kernel function to provide results with higher accuracy and performance. Minimal training error and improved approximation are the primary advantages of the ELM. Since ELM employs non-zero activation functions and the auto-tuning of weight biases, it finds application in classification and classification values. While the activation function of the output layer in this type of system is linear, the "L" neurons in the hidden layer must operate with a highly differentiable activation function (for example, the sigmoid function). The tuning of hidden layers in ELM is not required. The hidden layer in ELM does not necessarily need to be modified. It has a high training speed. Loads of the concealed layer are chosen at random (counting the bias loads). Hidden nodes are not insignificant, but they do not need to be tweaked, and the parameters of the hidden neurons may be produced randomly even before dealing with the training set data, which is advantageous. The formula for the system yield for a single-hidden layer ELM is:

$$f_L(x) = \sum_{i=1}^{L} \beta_i h_i(x) = h(x)\beta \tag{4}$$

where, x → input

β → Output weight vector and it is mathematically expressed as follows:

$$\beta = [\beta_1, \beta_2, \dots \dots \dots \dots . \beta_L]^T \tag{5}$$

H(x) → the mathematical expression for the output hidden layer looks like this:

$$h(x) = [h_1(x), h_2(x), \dots \dots \dots \dots .. h_L(x) \tag{6}$$

To find "Output vector O," also known as the "target vector," the mathematical expressions for the hidden layers are as follows:

$$H = \begin{bmatrix} h(x_1) \\ h(x_2) \\ \vdots \\ h(x_N) \end{bmatrix} \tag{7}$$

The minimal non-linear least square approaches significantly employed for the simple application of the ELM are represented in the following manner:

$$\beta' = H^{*}O = H^T(HH^T)^{-1}O \tag{8}$$

where, the H∗→ inverse of H is known as the Moore−Penrose generalized inverse.

The expression can alternatively be written as follows:

$$\beta' = H^T(\frac{1}{C}HH^T)^{-1}O \tag{9}$$

Using the preceding formulation, the output function is as follows:

$$f_L(x) = h(x)\beta = h(x)H^T(\frac{1}{C}HH^T)^{-1}O \tag{10}$$

3.5 Proposed architecture

A lightweight self-attention-based ensembled Convolutional Capsule network was designed using VGG- 1 6 topology. It consists of 1 6 layers and attention modules were embedded after the 14th layer.

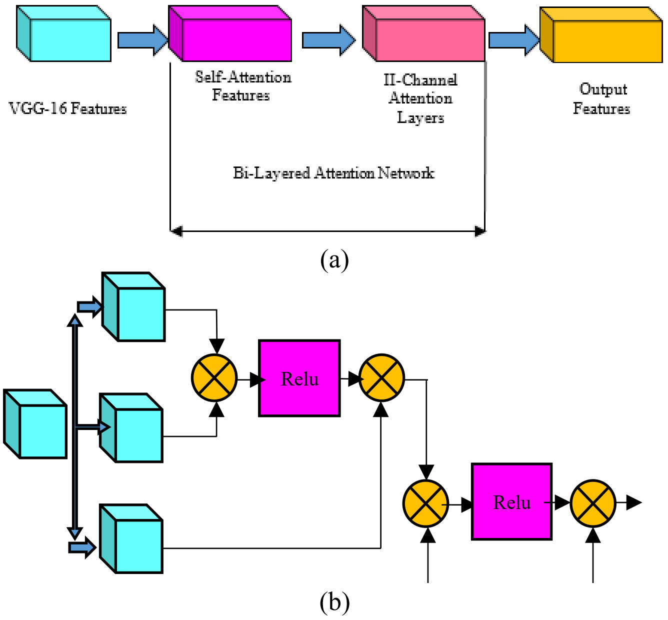
Inspired by the suggested approach, the Bi-layered Attention layers (BL- SAL) are implemented for the efficient extraction of multidimensional features. In the first layer, pipelined convolutional and ReLU activation functions are employed to extract intermediate features from the VGG-19 dataset. This significantly selects the relevant features. The initial convolutional layers employ element-wise multiplication, which is subsequently forwarded to the SoftMax layer to generate an attention map. To generate the first layered self-attention maps, the attention maps were multiplied by the transposition of the third layer's feature maps. The second layer, which comprises channel attention layers, receives the obtained self-attention maps and refines them. This approach reduces the total number of boundaries to make it lighter and portable by using kernel filters that are eight times lower for each base network layer than for standard CNN architectures or VGG-19. The suggested training network's block design is depicted in Figure 3, and Table 3 lists the training parameters that were utilized to build the proposed model.

$$\mathrm{B(F) = B'(F1) * B''(F2)} \tag{11}$$

$$\text{Where } \mathrm{B'(F1) = S(F1).\ V(F1)} \tag{12}$$

$$\mathrm{S(F1) = Soft(U^F).\ M(F)} \tag{13}$$

$$\mathrm{B''(F2) = B'(F1) * C(F2)} \tag{14}$$



(a)

(b)

Figure 3. a) Bi-layered attention networks integrated in the proposed architecture b) Generation of Bi-layered attention maps

Eq. (11) represents the refined attention features after passing to the bi-1ayered Attention maps. Eqs. (12), (13), and Eq. (14) presents the mathematical expressions performed using the bi-layered attention maps. Each value in the proposed attention maps represents the degree of attention paid to the corresponding feature vectors obtained in the phase. The low-level and high-level features are extracted using the BL-SAL maps. These features are then used to construct the complete capsule networks. As the features consist of both high and level attention maps, intermediate capsules are constructed, and then use all intermediate capsules to construct the primary capsules. The intermediate capsules are formulated using Eq. (ll) in which the spatial information is extracted between the high and low-level attention maps. These maps are then summed up using Eq. (12) to form the higher primary capsules. Additionally, a linear combination between the capsules is introduced to reduce the number of capsules at the same pixel location. Finally, the squashing function is applied using Eq. (13) on the linearly combined capsules so that the length is reduced to [0,1]. The positive side of the linear combination is that it reduces the redundant information present in the same pixel thus increasing the performance of classification. Finally, these capsules are used to train the feedforward layers which works on the principle of ELM using Eq. (10) to classify the multiple class diseases in plants.

Table 3. Parameters used for the modelling of the proposed architecture

Specifications	Parameters
VGG- 1 6 Networks	
Input Image Size	224×224×3
Conv layers	64×2
Pooling layer	Max-pooling layer
Conv layers	1 28×4
Pooling layers	Max-layers
Conv layers	256×4
Pooling layers	Max-pooling layers
Conv layers	5 1 2×4
Pooling layers	Max-pooling layers
Bi- Layered Attention layers	
Conv layers	5 12×3
Activation for 1 layer	Relu
Activation for 11-layer	Softmax
Classification layers	
Optimizer used	Adam
Activation Function	Rely
Normalization	Batch

4. SYSTEM SETUP

The proposed architecture has been implemented in the PC workstation with 17 CPUs with NVIDIA Tesla GPU, I TB SSD, and 3.2 GHZ operating system. Tensorflow 2.9.0 with Keras 2.9.0 is used for deploying the proposed network.

5. EVALUATION METRICS

The mathematical formulas used to evaluate the suggested approach are shown in Table 4. The metrics are examined and contrasted with those of other existing algorithms in terms of their accuracy, precision, recall, specificity, and F1 score such as CAPSNET [24], CAPS –SV [25], LWACNN [26] and transfer learning algorithms [27]. It is worth highlighting that the experiment's performance measures are based on the average of 5 simulated runs, with each consisting of different plant images. The training parameters used for experimentation are presented in Table 5.

Table 4. Mathematical expression for the performance metrics used for evaluation

Performance Metrics	Mathematical Expression
Accuracy	$\frac{TP+TN}{TP+TN+FP+FN}$
Recall	$\frac{\mathrm{TP}}{\mathrm{T\ P+FN}}$ x100
Specificity	$\frac{TN}{TN+FP}$
Precision	$\frac{TN}{TP+FP}$
F1-Score	$2.\frac{Precison * Recall}{Precision + Recall}$

Table 5. Training hyper-parameters used for the proposed model

Hyperparameters	Values
Batch Sizes	**35**
No of Epochs	200
Learning Rate	0.0001
Loss Function Employed	Cross-Entropy
Momentum for ADAM optimizer	0.1
Drop-out	0.2

6. RESULTS AND DISCUSSION

Table 6 demonstrates the normalized confusion metrics generated for the proposed model using different image datasets. The diagonal value shows the proportion of successfully identified plant disease images to all applied plant disease photos. Figure 4 shows the learning curves for the proposed architecture.

Table 6. Confusion matrix provided for proposed model in detecting the multiple diseases in a) Apple b) Blueberry c) Cherry d) Crape e) Orange O Peach g) Pepper bell h) Tomato

(a)

LABEL	Healthy	Apple scab	Black rot	Cedar Apple rust
Healthy	98.8	0	0	1
Apple Scab	0	99.45	0	1
Blackrot	1	0	99.4	0
Ceder Apple xrust	0	1	0	99.56

(b)

LABEL	Healthy	Powdery mildew
Healthy	99.8	0
Powdery mildew	0	99.7

(c)

LABEL	Healthy	Powdery mildew
Healthy	99.78	0
Powdery mildew	0	99.7

(d)

LABEL	Healthy	Black Rot	Esca Black Measles
Healthy	98.8	0	1.1
Black Rot	0	98.86	1.1
Esca black measles	1.1	0	98.86

(e)

LABEL	Healthy	Powdery mildew
Healthy	99.78	0
Powdery mildew	0	99.7

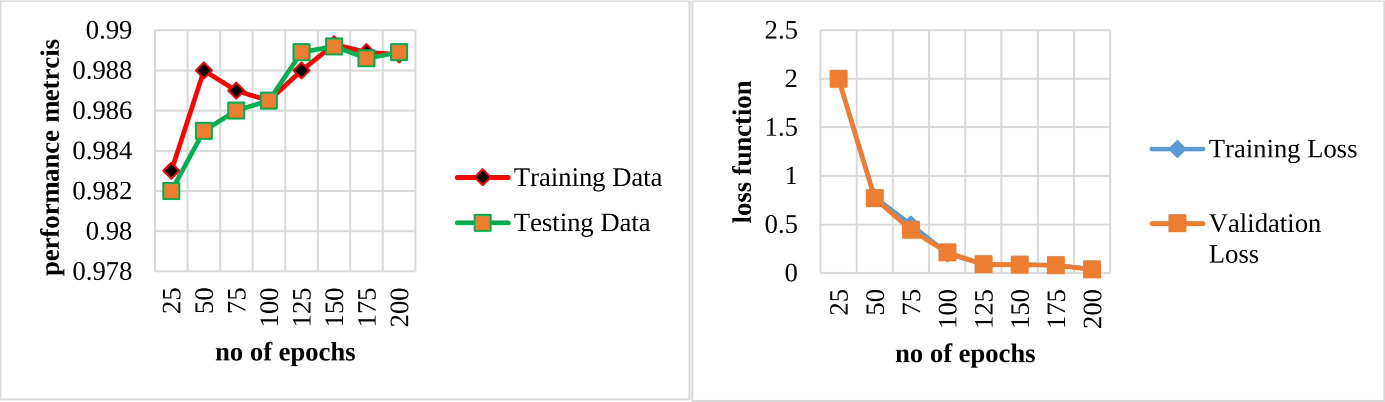
(f)

LABEL	Healthy	Powdery mildew
Healthy	99.7	0
Powdery mildew	0	99.78

(g)

LABEL	Healthy	Early blight	Late blight
Healthy	99.92	0	0
Early blight	0	99.96	0
Late blight	1	0	99.92

Tables 7-14 present the proposed algorithm's performance in detecting multiple-scale diseases from multiple plants. The proposed model was evaluated using testing data with 14 samples and 15000 distant and multiple image datasets. The chart indicates that the proposed framework has provided an average performance of 99.8 percent accuracy, 99.75 percent precision, 99.65 percent recall, and 99.75 percent F I score. The model with the Capsule and Attention model has produced the maximum performance in detecting the multiple scales of diseases from multiple plants.



(a) **(b)**

Figure 4. Learning curves for the proposed model a) Performance learning curves b) Loss learning curves

Table 7. Proposed model's validation metrics for detecting the diseases in apple plant

Plant and Disease Type	Performance Metrics				
	Accuracy (%)	Precision (%)	Recall (%)	Specificity (%)	F1-Score (%)
Healthy Apple	99.6	99.5	99.35	0.011	99.4
Apple Scab	99.56	99.45	99.5	0.015	99.5
Blackrot	99.62	99.5	99.35	0.011	99.4
Ceder Apple xrust	99.9	99.8	100	0.01	99.89
Average Performance	**99.67**	**99.6**	**99.7**	**0.011**	**99.78**

Table 8. Proposed model's validation metrics model for detecting the diseases in strawberry plant

Plant and Disease Type	Performance Metrics				
	Accuracy (%)	Precision (%)	Recall (%)	Specificity (%)	F1-Score (%)
Healthy Strawberry	99.9	99.7	99.5	0.011	99.4
Unhealthy Scab	100	99.8	100	0.01	99.89
Average Performance	**99.9**	**99.75**	**99.0**	**0.011**	**99.6**

Table 9. Proposed model's validation metrics for detecting the diseases in corn plant

Plant and Disease Type	Performance Metrics				
	Accuracy (%)	Precision (%)	Recall (%)	Specificity (%)	F1-Score (%)
Healthy Corn	99.9	99.7	99.5	0.011	99.4
Unhealthy Scab	100	99.8	100	0.01	99.89
Average Performance	**99.9**	**99.75**	**99.0**	**0.011**	**99.6**

Table 10. Proposed model's validation metrics for detecting the diseases in squash plant

Plant and Disease Type	Performance Metrics				
	Accuracy (%)	Precision (%)	Recall (%)	Specificity (%)	F1-Score (%)
Healthy Squash	99.9	99.7	99.5	0.011	99.4
Unhealthy Scab	100	99.8	100	0.01	99.89
Average Performance	99.9	99.75	99.0	0.011	99.6

Table 11. Proposed model's validation metrics for detecting the diseases in raspberry plant

Plant and Disease Type	Performance Metrics				
	Accuracy (%)	Precision (%)	Recall (%)	Specificity (%)	F1-Score (%)
Healthy Raspberry	99.9	99.7	99.5	0.011	99.4
Unhealthy Scab	100	99.8	100	0.01	99.89
Average Performance	99.9	99.75	99.0	0.011	99.6

Table 12. Proposed model's validation metrics for detecting the diseases in soyabean plant

Plant and Disease Type	Performance Metrics				
	Accuracy (%)	Precision (%)	Recall (%)	Specificity (%)	F1-Score (%)
Healthy Soyabean	99.9	99.7	99.5	0.011	99.4
Unhealthy Scab	100	99.8	100	0.01	99.89
Average Performance	99.9	99.75	99.0	0.011	99.6

Table 13. Proposed model's validation metrics for detecting the diseases in potato plant

Plant and Disease Type	Performance Metrics				
	Accuracy (%)	Precision (%)	Recall (%)	Specificity (%)	F1-Score (%)
Healthy Potato	99.84	99.76	99.63	0.012	99.7
Light Blight	100	99.8	100	0.01	99.89
Early Blight	100	99.75	99.78	0.01	99.77
Average Performance	99.4	99.77	99.80	0.01	99.78

Table 14. Proposed model's validation metrics for detecting the diseases in tomato plant

Plant and Disease Type	Performance Metrics				
	Accuracy (%)	Precision (%)	Recall (%)	Specificity (%)	F1-Score (%)
Healthy Tomato	99.6	99.5	99.35	0.011	99.4
Light Blight	99.56	99.45	99.5	0.015	99.5
Early Blight	99.62	99.5	99.35	0.011	99.4
Bacterial Spot	99.9	99.8	100	0.015	99.89
Leaf Spot	99.9	99.8	99.76	0.023	99.78
Target Spot	99.8	99.7	99.8	0.02	99.76
Yellow Leaf Virus	99.87	99.83	99.75	0.012	99.78
Mosoic Virus	99.0	99.8	100	0.01	99.89
Spider Mates	100	99.56	99.74	0.011	99.67
Average Performance	99.86	99.75	99.7	0.0112	99.78

6.1 Comparative analysis

To demonstrate the superiority of the proposed framework, the performance of its deep transfer learning and capsule network counterparts is compared.

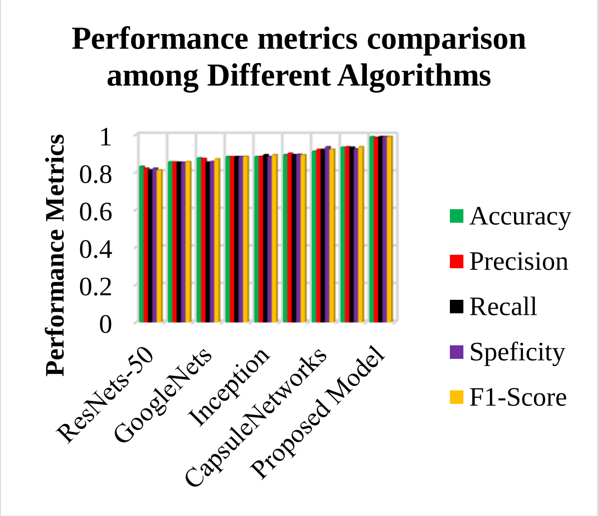


Figure 5. Comparison of the average efficacy of various algorithms for spotting illnesses in healthy plants

Figure 5 and Figure 6 represent the average performance of the various models in recognizing healthy and unhealthy plant diseases from multiple-scale plant images. From Figures 5 and 6, the traditional CNN learning techniques have produced the lowest performance in detecting multiple plant diseases. The CAPSNET and hybrid CAPSNET have considerably produced better performance than the traditional learning techniques. Notably, the inclusion of attention networks in the LWATTENTION Model and the proposed model has produced significantly better performance than the other algorithms. In the same scenario, other frameworks degrade in their performance because they do not support the large-scale datasets have high computational complexities, and require more time for training. However, the integration of capsule networks along with bi-layered attention maps has outperformed lightweight attention networks in detecting the multiple scales of diseases from different plants due to its novel architecture. From the results, it is observed that the proposed framework has produced a better average performance in multiple scales of plant diseases and can be utilized by the farmers to achieve better yield and cultivation.

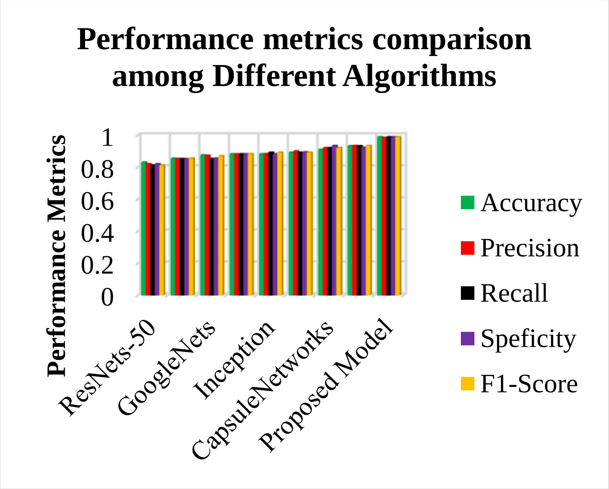


Figure 6. Comparative evaluation of the average effectiveness of various algorithms for identifying plant diseases

7. CONCLUSION

The unique ensembling of Capsule networks with bi-layered attention maps was tested in this study to find a way to identify many scale plant illnesses. The bi-layered attention network and capsule networks are proposed to separate the fundamental distinguishing characteristics of the various plants. Additionally, ELM-based Feedforward layers are coupled to the discriminant capsules to achieve better performance of classification. The proposed network architecture and performance were analyzed in terms of accuracy, precision, recall, specificity, and F1 score and compared with the other learning frameworks. Experimental results show that the proposed framework effectively extracts essential features from the multiple scale of plants in turn achieving better performances than the other prevailing deep and attention models. The results reveal that the proposed architecture has achieved 99.8 percent accuracy, 99.75 percent precision, 99.65 percent recall, and 99.75 percent F1- score. Hence the proposed framework achieved better classification performance in plant disease identification. As for future scope, the proposed model should be refined in terms of implementation in real-time loT (Internet of Things) devices. Also, the analysis can be performed at the cloud platforms.

REFERENCES

[1] Liu, J., Wang, X. (2020). Tomato diseases and pests detection based on improved Yolo V3 convolutional neural network. Frontiers in Plant Science, 11: 898. https://doi.org/10.3389/fpls.2020.00898

[2] Wang, G., Sun, Y., Wang, J. (2017). Automatic image-based plant disease severity estimation using deep learning. Computational Intelligence and Neuroscience, 2017: 2917536. https://doi.org/10.1155/2017/2917536

[3] Food and Agriculture Organization. (2008). Climate Related Transboundary Pests and Diseases. FAO: Rome, Italy, p. 59.

[4] Fukushima, K. (1980). Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. Biological Cybernetics, 36(4): 193-202. https://doi.org/10.1007/BF00344251

[5] LeCun, Y., Boser, B., Denker, J.S., Henderson, D., Howard, R.E., Hubbard, W., Jackel, L.D. (1989). Backpropagation applied to handwritten zip code recognition. Neural Computation, 1(4): 541-551. https://doi.org/10.1162/neco.1989.1.4.541

[6] Kern, A., Barcza, Z., Marjanović, H., Árendás, T., Fodor, N., Bónis, P., Bognár, P., Lichtenberger, J. (2018). Statistical modelling of crop yield in Central Europe using climate data and remote sensing vegetation indices. Agricultural and Forest Meteorology, 260: 300-320. https://doi.org/10.1016/j.agrformet.2018.06.009

[7] Son, N.T., Chen, C.F., Chen, C.R., Guo, H.Y. (2020). Classification of multitemporal Sentinel-2 data for field-level monitoring of rice cropping practices in Taiwan. Advances in Space Research, 65(8): 1910-1921. https://doi.org/10.1016/j.asr.2020.01.028

[8] Zhang, H., Kang, J., Xu, X., Zhang, L. (2020). Accessing the temporal and spectral features in crop type mapping using multi-temporal Sentinel-2 imagery: A case study of Yi'an County, Heilongjiang province, China. Computers and Electronics in Agriculture, 176: 105618. https://doi.org/10.1016/j.compag.2020.105618

[9] Dey, S., Mandal, D., Robertson, L.D., Banerjee, B., Kumar, V., McNairn, H., Bhattacharya, A., Rao, Y.S. (2020). In-season crop classification using elements of the Kennaugh matrix derived from polarimetric RADARSAT-2 SAR data. International Journal of Applied Earth Observation and Geoinformation, 88: 102059. https://doi.org/10.1016/j.jag.2020.102059

[10] Planque, C., Lucas, R., Punalekar, S., Chognard, S., Hurford, C., Owers, C., Horton, C., Guest, P., King, S., Williams, S., Bunting, P. (2021). National crop mapping using sentinel-1 time series: A knowledge-based descriptive algorithm. Remote Sensing, 13(5): 846. https://doi.org/10.3390/rs13050846

[11] Azadbakht, M., Ashourloo, D., Aghighi, H., Radiom, S., Alimohammadi, A. (2019). Wheat leaf rust detection at canopy scale under different LAI levels using machine learning techniques. Computers and Electronics in Agriculture, 156: 119-128. https://doi.org/10.1016/j.compag.2018.11.016

[12] Park, K., ki Hong, Y., hwan Kim, G., Lee, J. (2018). Classification of apple leaf conditions in hyper-spectral images for diagnosis of Marssonina blotch using mRMR and deep neural network. Computers and Electronics in Agriculture, 148: 179-187. https://doi.org/10.1016/j.compag.2018.02.025

[13] Karadağ, K., Tenekeci, M.E., Taşaltın, R., Bilgili, A. (2020). Detection of pepper fusarium disease using machine learning algorithms based on spectral reflectance. Sustainable Computing: Informatics and Systems, 28: 100299. https://doi.org/10.1016/j.suscom.2019.01.001

[14] Iqbal, Z., Khan, M.A., Sharif, M., Shah, J.H., ur Rehman, M.H., Javed, K. (2018). An automated detection and classification of citrus plant diseases using image processing techniques: A review. Computers and Electronics in Agriculture, 153: 12-32. https://doi.org/10.1016/j.compag.2018.07.032

[15] Singh, A.K., Kumar, S., Bhushan, S., Kumar, P., Vashishtha, A. (2021). A proportional sentiment analysis of MOOCs course reviews using supervised learning algorithms. Ingénierie des Systèmes d'Information, 26(5): 501-506. https://doi.org/10.18280/isi.260510

[16] Kumar, S.P., Anandan, R. (2022). A comprehensive analysis on secured data storage with user validation and resource allocation in cloud for performance enhancement. International Journal of Safety and Security Engineering, 12(2): 137-144. https://doi.org/10.18280/ijsse.120201

[17] Zhang, Y., Song, C., Zhang, D. (2020). Deep learning-based object detection improvement for tomato disease. IEEE Access, 8: 56607-56614. https://doi.org/10.1109/ACCESS.2020.2982456

[18] Verma, S., Chug, A., Singh, A.P. (2020). Exploring capsule networks for disease classification in plants. Journal of Statistics and Management Systems, 23(2): 307-315. https://doi.org/10.1080/09720510.2020.1724628

[19] Peker, M. (2021). Multi-channel capsule network ensemble for plant disease detection. SN Applied Sciences, 3(7): 707. https://doi.org/10.1007/s42452-021-04694-2

[20] Altan, G. (2020). Performance evaluation of capsule

networks for classification of plant leaf diseases. International Journal of Applied Mathematics Electronics and Computers, 8(3): 57-63. https://doi.org/10.18100/ijamec.797392

[21] Waweru, L.W., Kipyego, B.T., Muchangi, D.M. (2021). Classification of plant leaf diseases based on capsule network-support vector machine model. International Journal of Electrical Engineering and Technology, 12(6): 188-199. https://doi.org/10.34218/IJEET.12.6.2021.018

[22] Pei, S.C., Hsiao, Y.Z. (2015). Spatial Affine transformations of images by using fractional shift fourier transform. In 2015 IEEE International Symposium on Circuits and Systems (ISCAS), Lisbon, Portugal, pp. 1586-1589. https://doi.org/10.1109/ISCAS.2015.7168951

[23] Domathoti, B., Ch, C., Madala, S., Berhanu, A.A., Rao, Y.N. (2022). Simulation analysis of 4G/5G OFDM systems by optimal wavelets with BPSK modulator. Journal of Sensors, 2022: 8070428. https://doi.org/10.1155/2022/8070428

[24] Mazzia, V., Salvetti, F., Chiaberge, M. (2021). Efficient-capsnet: Capsule network with self-attention routing. Scientific Reports, 11(1): 14634. https://doi.org/10.1038/s41598-021-93977-0

[25] Mahankali, S., Kalava, J., Garapati, Y., Domathoti, B., Sundramurty, V.P. (2022). A treatment to cure diabetes using plant-based drug discovery. Evidence-Based Complementary and Alternative Medicine, 2022: 8621665. https://doi.org/10.1155/2022/8621665

[26] Zhang, K., Wu, Q., Liu, A., Meng, X. (2018). Can deep learning identify tomato leaf disease? Advances in Multimedia, 2018: 6710865. https://doi.org/10.1155/2018/6710865

[27] Abbas, A., Jain, S., Gour, M., Vankudothu, S. (2021). Tomato plant disease detection using transfer learning with C-GAN synthetic images. Computers and Electronics in Agriculture, 187: 106279. https://doi.org/10.1016/j.compag.2021.106279