International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 10s
DOI: https://doi.org/10.17762/ijritcc.v11i10s.7695
***Article Received:** 08 June 2023 **Revised:** 29 July 2023 **Accepted:** 14 August 2023*

FLY-CAPS- A Hybrid Firefly Feature Optimized Capsule Networks for Plant Disease Classification in Resource Constriant Internet of Things (IoT)

Ponugoti Kalpana[1], R. Anandan[2]
[1]Research Scholar, Department of Computer Science and Engineering,
Vels Institute of Science, Technology and Advanced Studies,
Chennai 600117, Tamil Nadu,India.
Email: kalpanaraogonait@gmail.com
[2]Professor, Department of Computer Science and Engineering,
Vels Institute of Science, Technology and Advanced Studies,
Chennai 600117, Tamil Nadu, India.
Email: anandan.se@velsuniv.ac.in

Abstract—Recent advancements in artificial intelligence, automation, and the Internet of Things (IoT) enable farmers to better monitor and diagnose all agricultural procedures with super-intellectual accuracy. These technologies also contribute to boosting the productivity of agriculture, which increases the country's economy. Though these technologies help farmers increase productivity, the detection of plant diseases still needs heightened scrutiny for prevention and cultivation. Plant disease categorization has expanded with the introduction of deep learning algorithms, but it still needs more innovation in terms of accuracy and computing burden. Thus, a novel deep learning model based on capsule networks with firefly optimization and potent multi-layered feedforward prediction networks is proposed in this research. The handcrafted features in this proposed system are optimized before being extracted using a capsule network, which reduces the complexity overhead and is suitable for IoT devices with limited resources. Finally fed to the feed forward layers for better classification. The extensive experimentation has been tested with the Plant Village databases, which contain more than 50,000 images of healthy and infected plants. Performance criteria including recall, specificity, recall, accuracy, and f1-score are used to assess the proposed algorithm's performance. Additionally, its efficiency and computational cost are contrasted with those of other recent models. The suggested model has greater performance (95%) with reduced computing overhead, according to experimental data, which is advantageous for the new prediction approach and the welfare of the farmer.

Keywords- Artificial Intelligence; the Internet of Things (IoT); Capsule Networks; Firefly Optimization; Multi-layered Feedforward layers;

I. INTRODUCTION

Agriculture has been central to the human race's survival, development, and existence, as it is the primary source of their daily food. Even today, agriculture plays an important role in every nation's GDP and provides nearly 50% of the total employment in the country. [1] Many factors, such as climate change, artificial fertilizers, and a lack of manpower, have already affected the agricultural yield across the globe and are still beyond our control. In addition to the problems mentioned above, plant diseases are regarded as one of the most dangerous economic damages because they affect both food quality and yield production. [2].

These plant diseases are being identified using a variety of techniques and procedures. Nevertheless, some farmers are fortunate enough to afford labs and professional oversight. The majority of them lack the resources to pay for experts and tools that would enable them to identify plant diseases in their early stages and stop their spread. Farmers must also take the sample to the certified labs to diagnose the ailment in the plant. These labs' setup cost a lot of money and resources, and those resources would be better invested in other projects that would increase production. [3] In order to prevent such sickness, agricultural ecosystems that are complex and cognitive must be able to continuously monitor and anticipate such dangers [4].

To process images and analyze data effectively, machine learning and deep learning algorithms can be used. These cutting-edge methods have a lot of potential and consistently provide promising results. These algorithms can be used to create a unique ecosystem that can accurately anticipate plant illnesses. Machine learning predicts plant diseases using handcrafted features, which increases in complexity over time and has an impact on prediction performance. Presently, due to the features of deep learning algorithms, many researchers are using them for the effective prediction of plant diseases. Convolutional neural networks (CNN [5–

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 10s
DOI: https://doi.org/10.17762/ijritcc.v11i10s.7695
***Article Received:** 08 June 2023 **Revised:** 29 July 2023 **Accepted:** 14 August 2023*

9]) are a family of deep learning models that are used for efficient feature extraction and dimensionality reduction in the data and are utilised in contrast to other learning models for accurate classification of plant diseases. But when the quantity of datasets increases, the computational complexity of CNN also increases. Therefore, to obtain the more encouraging results in the classification of plant diseases, the CNN has been built utilising pretrained models such as Google Nets, Dense Net, and Mobile Net [10–12]. However, these methods need improvisation in terms of their low complexity and high false alarm rates. Additionally, these algorithms may fail to train on the larger crop datasets.

II. MOTIVATION:

This research suggests an innovative Firefly feature optimization assembly with capsule networks and multi-layered feedforward networks to handle the larger crop datasets with greater false alarm rates and less computing complexity in response to the constraints. To the best of our knowledge, this is the first of its kind in assembling the different layers of learning models to achieve the three important objectives of low complexity, a low false alarm rate, and handling larger datasets**.**

1. CONTRIBUTION OF THE RESEARCH: The paper's primary contribution is as follows: 1. To choose features from capsules and build a lightweight network appropriate for the Internet of Things, the study suggests a hybrid feature optimization strategy based on the Firefly algorithm. This study also employs a capsule layer, which enables the system to record the spatial and angular relationships between various disease entities in a picture.
2. Multiple performance indicators are generated and compared with those of other cutting-edge deep learning models as part of a comprehensive experimentation process that validates and assesses the proposed method.

ORGANIZATION OF THE PAPER: The paper is structured as follows:

THE WORKS RELATED TO SECTION II:

Machine Learning and Hybrid model for plant disease detection

A method for recognising and classifying diseases that affect weeds and crops was proposed by Le et al. First, the noise in the input photos was eliminated using morphological opening and closing approaches. A modified framework known as the filtered local binary pattern technique with contour mask and coefficient k was utilised to determine the features from the processed sample (k-FLBPCM). The SVM classifier was trained to categorise a range of plant diseases using the attributes that were acquired. The approach outlined in [13] improves the classification accuracy of plant diseases, however it may not be effective with samples that exhibit perspective distortion.

Pathogens that harm tea plants may have a hierarchy, according to Sun et al[14] .'s hypotheses. The Harris technique was used to extract the main points from an input sample after it had been divided into blocks using the "Simple Linear Iterative Cluster (SLIC)". The fuzzy salient region contour was constructed using the convex hull technique, and the key points were extracted using the "Gray Level Co-occurrence Matrix (GLCM)" method. The SVM classifier was then taught to recognise the ailments affecting the leaves of the tea plant. The approach provides a higher classification accuracy, but at a high computational cost. Ramesh as well as others [15] suggested a system for categorising plant diseases. The "Random Forest (RF)" model was trained to use these characteristics to identify the data into normal and infected groups using "Histogram of Oriented Gradients (HOGs)" features. The approach is strong in terms of crop disease categorization, but its performance might be better.

A classification system for disorders affecting turmeric leaves was provided by Kuricheti et al. [16]. After pre-processing, the K means approach was used to perform edge detection on the input image. Then, features were extracted using the GLCM method, and the SVM method was trained to categorise the leaves. Although this method yields superior results for identifying plant diseases, it is unable to outperform other methods on samples with significant brightness changes.

Deep Learning and Hybrid model for plant disease detection

Few-Shot Learning, created by Arguesoa et al. [17], is a DL-based approach for identifying and categorising fungal diseases (FSL). The Inception V3 framework was first used to determine the key points. An SVM with several classes was then fed the features that had been gathered (SVM). Although this method is resistant to categorization of agricultural diseases, the results are displayed on a small dataset and must be verified on a large and varied corpus. A CNN-based architecture was created by P. T. Miller et al. [18] to localise and classify the tomato crop sickness. This method uses three convolutions and max-pooling layers to identify and categorise the key points in the input samples. Although this method is more accurate in classifying tomato diseases, it has the drawback of being overfit over a limited number of classes.

According to Richey et al. [19], a method has been developed for identifying and classifying illnesses that impact maize crops using a mobile app. To compute the deep key points from the input photographs and categorise them, a DL-based model dubbed "ResNet50" was trained over the ImageNet database. The method offers a smartphone-based approach for classifying agricultural diseases, however due to memory, processing, and battery power limitations, it is computationally intensive and not well suited for smartphones. A brand-new DL-based categorization system for diseases affecting tomato crops was presented by Zhang et al. [20]. In their customised Faster-RCNN technique, the researchers suggested using the deep residual framework to extract data rather than the VGG16 model. Furthermore, the edge pixel data were grouped using the k-means clustering method. Although it has a more financial impact, this technology enhances disease identification findings for tomato crops.

Early on, a method for recognising and categorising tomato leaf disease was devised by Batool et al. To train the "KNN" to classify the photos as healthy or impacted, deep feature points were first collected from the input pattern using the "Alex Net" platform. KNN improves prediction performance, however it is a slow and time-consuming technology. Karthik et al. [22] developed a DL-based approach to identify illness in tomato leaves. The deep properties of the input sequence were computed using a residual network. After that, a CNN model was trained to produce critical points that might be used to distinguish between healthy and sick leaves. Despite being more accurate in classifying leaf diseases, this technique is not economical.

The deep key points of diverse plants were calculated using various DL-based models, such as "Alex Net, Google Net, DenseNet201, ResNet50, and ResNet101" frameworks, according to an ensemble technique proposed by Turkoglu et al. [23]. Following that, SVM training was done on the calculated attributes to classify various plant diseases. The approach does, however, raise plant leaf classification performance at the expense of more expensive feature calculation.

Capsule network-based plant disease classification

In [24], Shradha Verma et al investigated capsule networks for identifying disinfected plants. The author adopted the standard capsule network, which plays a significant role in the agricultural field in terms of "texture, orientation, and pose" more accurately than deep learning algorithms. Capsule network architecture collects each and 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 has not been validated with segmented or real-time datasets.

Loise Wanjiru et al adopted the capsule network for classifying plant diseases in [25]. For the categorization of tomato leaf diseases, a new model based on the integration of Caps Net and support vector machines (Caps Net-SVM) was investigated. The SVM model was utilised as a robust classifier, while the capsule network was optimised for feature extraction. The major goal was to improve support vector machine classification by employing artificial features retrieved by the capsule network model. The Caps Net-SVM model was found to be capable of autonomously extracting features from raw photos and performing final classification.

G. Altan et al The "Caps NET architecture" was introduced to investigate the efficiency of the algorithm in the detection of plant leaf diseases using decreased capsules on leaf photos. Plant leaf diseases are prevalent and common illnesses that have a disastrous impact on agricultural production and harvests. Even little stains that may have an impact on the timing and duration of seed dressing can be thoroughly analysed using Caps NET. The proposed "Caps NET" model aimed to enhance the learning capability of DL models and examine the applicability of various feature learning techniques for bell pepper crops. Images of healthy and ill leaves were sent to Caps NET [26].

Kwabena et al suggests using the Gabor and Capsule networks to identify photos of tomato and citrus diseases that are blurred, distorted, and hidden [27]. With its filters restricted to fit a Gabor function, the convolutional layer used to implement the Gabor layer. Compared to the CNN model, the architecture complexity is relatively high.

M. Peker et al. [28] proposed the ensemble capsule network with five channels for plant disease detection. To achieve the best performance in detecting the texture, shape, or leaf affected area more accurately, Gabor, PCA, and filter networks were combined with a capsule network. The limitation of this network is its high training complexity, and overfitting problems arise at the pre-processing level.

SECTION-III

III. PROPOSED METHODOLOGY:

3.1 SYSTEM OVERVIEW

A hybrid and efficient model called FLY-CAPNETS is suggested to predict plant illnesses from the gathered photos. FLY-CAPNETS combines the three separate learning layers to produce a highly accurate prediction of

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 10s
DOI: https://doi.org/10.17762/ijritcc.v11i10s.7695
***Article Received:** 08 June 2023 **Revised:** 29 July 2023 **Accepted:** 14 August 2023*

plant diseases. The overall pipelined architecture is shown in Figure 1. The three learning layers along with their applications used in this research are tabulated in Table 1

Table 1 List of the Ensembled Layers used in the TEN-CAPNETS

SL.NO	Learning Layers	Advantages
01	VGG-19 Transfer learning	Feature Extraction mechanism
02	Capsule networks	
03	Multi-layered Feed Forward layers	Better prediction of plant diseases.

The VGG-19 learning network serves as the feature extractor unit in the suggested model. The model employs transfer learning and a state-of-the-art Capsule network in place of the last layer to extract the relative spatial and orientational link between the numerous image entities of a plant. Additionally, feedforward layers rather than traditional thick layers are used for better complexity prediction. Another way the suggested model learns is through the ensemble networks, which replace a pipeline of components and go straight from the input to the desired output. By learning directly from labelled data, this technique eliminates the need for a time-consuming feature extraction phase and speeds up decision-making. End-to-end models are essential for creating artificial intelligence systems due to their effectiveness, efficiency, and data-driven nature. The following subsection discusses the materials used for training, transfer learning, capsule networks, and finally feedforward layers.

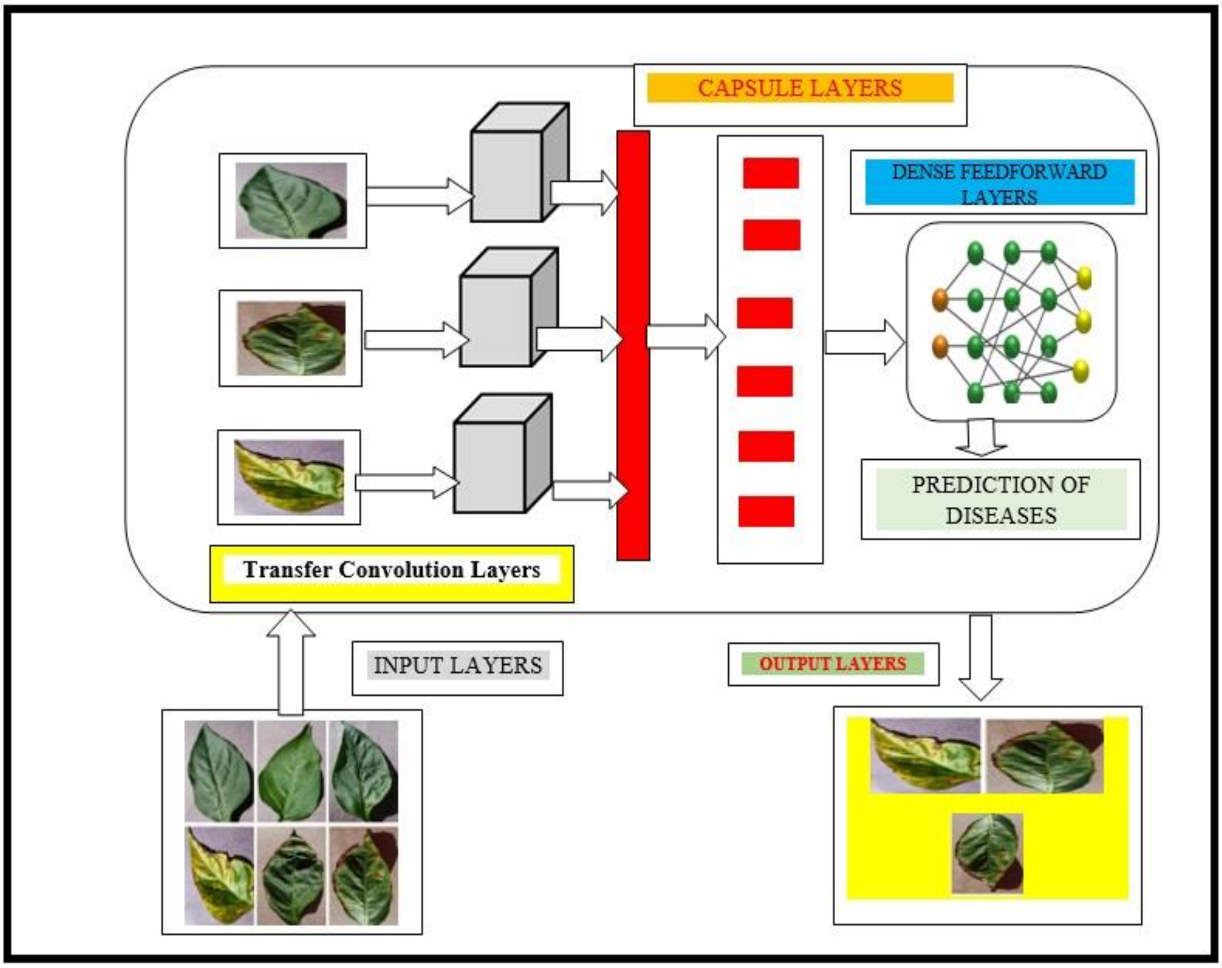

Figure 1 shows the proposed architecture's overall framework.

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 10s
DOI: https://doi.org/10.17762/ijritcc.v11i10s.7695
***Article Received:** 08 June 2023 **Revised:** 29 July 2023 **Accepted:** 14 August 2023*

3.2 MATERIALS AND METHODOLOGIES:

Plant Village, an open-access repository of photos on plant health acquired from a source [29], has been used for training and testing in this work to help with the development of mobile disease diagnosis. The Plant Village dataset's 54,306 images show 14 different plants. There are a total of 38 classes, of which 12 exhibit various plants with healthy leaves and 26 show various plant illnesses. Table 2 provides a description of the entire dataset. Figure 2 provides an illustration of samples of both healthy and sick plants.

Table 2: Representation of dataset

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 Mould	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	Haunglonbing	26	1201
Peach	Bacterial Spot	27	902
	Healthy	28	
Pepper bell	Bacterial Spot	29	2503
	Healthy	30	
Potato	Early blight	31	12901
	Healthy	32	
	Late Blight	33	
Raspberry	Healthy	34	1290
Soyabeans	Healthy	35	890\
Squash	Powdery Mildew	36	5690
	Healthy	37	
	Leaf Scorch	38	

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 10s
DOI: https://doi.org/10.17762/ijritcc.v11i10s.7695
***Article Received:** 08 June 2023 **Revised:** 29 July 2023 **Accepted:** 14 August 2023*


(a)


(b)


(c)


(d)

Figure 2 : Visual Representation of Health and Illness in a) healthy apple, b) pepper bell with bacterial spot, c) an apple with black rot, d) a tomato plant are examples of plants**.**

3.3 FEATURE EXTRACTION UNIT-ENSEMBLED FEATURE EXTRACTION AND CAPSULE NETWORK:

The ensemble of the pretrained model and capsule networks is covered in this section. The research's models are listed as follows:

3.3.1 CAPSULE NETWORKS -AN OVERVIEW:

Capsule networks [30] have recently been proposed as a remedy for this issue with existing networks. The encoding of geographic information and the propensity for an object to be present is carried out by capsules, which are collections of neurons. The network of capsules has a capsule that contains details about each entity in an image:

1. Entity existence probabilities

2. Parameters for Entities' Instantiation.

To transmit the essential spatial relationship between low-level and high-level elements inside the image, the input vectors' matrix is multiplied by the weight matrix. $Y(i.j) = W_{i,j}\, U(i,j) * S_j$ (1)

The weighted input vectors are added to determine which higher level capsule the current capsule will forward its output to. $S(j) = \sum_j Y(i,j) * D(j)$ (2)

Ultimately, non-linearity is applied using the squash function. The squashing function maintains the orientation of a vector while reducing its length to a range between 0 and 1. $G(j) = squash(S(j)$ (3)

3.2 PROPOSED FEATURE EXTRACTION:

The network receives 3-megapixel plant pictures with dimensions of 224 by 224. The input size of the suggested network is written as (y; 224 224 3), where y is the batch size. When inputs are fed into the capsule network, features or representations are extracted from them by passing them through four layers of convolution, LeakyRelu activation, and a max-pooling layer after each layer. The final layer of features is the feed-to-capsule network, which extracts the relative spatial and temporal aspects. By feeding these features to the firefly optimised flatten layer, the features are subsequently selected from the capsule networks.

3.3 FEATURE OPTIMIZED FLATTEN LAYERS:

In these flattened, feature-optimised layers, the firefly algorithm is adopted for the selection of the better features from the capsule networks and also reduces the complexity of the proposed algorithm. The detailed description of the

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 10s
DOI: https://doi.org/10.17762/ijritcc.v11i10s.7695
***Article Received:** 08 June 2023 **Revised:** 29 July 2023 **Accepted:** 14 August 2023*

Firefly optimization algorithms is discussed in the preceding section.

3.3.1 FIREFLY OPTIMIZATION ALGORITHM:

The Firefly algorithm, which is recognized as a member of the swarm intelligence algorithm family, was developed by Yang [31]. It's usual to see fireflies, sometimes known as lighting bugs, flashing their lights in the sky during the summer. Fireflies flash to either entice a possible mate or to ward off predators. Another important characteristic of fireflies is the way their light weakens as I get farther away from the brighter one and the way the air also influences this light's strength by absorbing it as the gap widens. As a result, there is a clear connection between light intensity and fitness value. However, in order to develop an algorithmic working principle, we must first build three hypotheses due to the intricacy of firefly natural behavior. The following assumptions are made:

1. It was supposed that all fireflies were unisex and that all of them experienced attraction.
2. The attractiveness of fireflies is inversely correlated with their brightness, and it decreases with increasing separation.
3. The practical solutions of the objective function are used to calculate the brightness or light intensity.

The hypotheses make it clearly clear that the light intensity of fireflies, I(r), is inversely related to distance r since it decreases with increasing distance and, once again, light is also absorbed while it travels through the air. The letter y stands for the coefficient of light absorption. The variation in firefly light intensity I(r) with respect to distance r is shown by equation (4)..

$$I(r) = I_0 e^{-yr2} \qquad (4)$$

where I0 represents the source's starting intensity value, and the attractiveness parameter can be defined in one of two ways as illustrated in
$\beta(r) = \beta 0 e^{-yr2}$ (5)

In zero at the beginning, attractive parameters are written as 0.

The equation below provides the behavioral rule for computing firefly positions.
$x_{i+1} = x_i + \beta(r(i,j))(xj - xi) + AE$ (6)

When E is the random number vector and A is the randomization factor, the Gaussian distribution is used to derive both variables. The value of attraction is represented by the xi+1second term, where xI is the ith position of the firefly.

Firefly attraction qualities like those indicated in (4), (5), and (6) are employed to improve feature extraction. The fitness function is intended as a multi-objective problem with the goal of reducing computational complexity and improving performance, as was indicated in [31].

We have considered feature selection as a multi-objective optimization function with two objective functions: low feature count and high classification accuracy, in order to obtain the best results. For each iteration, the fitness function is thus provided as follows.

$$\text{Fitness Function} = \mu\alpha(A) + \beta(\tfrac{S}{N}) \qquad (14)$$

The function represents accurate categorization (A). N stands for the overall number of features, and S for the feature vectors' multilinearity. The primary function that indicates classification accuracy is, while the length of feature subsets is.

3.4.1 FEEDFORWARD LAYERS:

The final layer of the suggested model swaps out dense neural network categorization for G.B. Huang's quick, powerful learning tools [32]. An ELM, a particular kind of neural network, has just one hidden layer and relies on the concept of auto-tuning features. The ELM has superior performance, fast speed, and computing efficiency when compared to other learning models like Support vector machines (SVM), Bayesian Classifier (BC), K-nearest neighbours (KNN), and even Random Forest (RF).

This neural network has a single hidden layer, which does not necessarily need to be tuned. ELM displays superior performance, high speed, and less computational overhead when compared to other learning algorithms like Support vector machines (SVM) and Random Forest (RF). ELM uses the kernel function to produce good accuracy for the better performance. Minimal training error and better approximation are the ELM's main benefits. Since ELM employs non-zero activation functions and the auto-tuning of weight biases. The ELM's precise working mechanism is described in depth in [33]. The ELM's input features mappings (following Capsule Network) are represented by

$$X = F(P) \qquad (4)$$

Where P is the features from the various types of capsule networks and X is the information from the Transfer Capsule network.

The symbol for the output ELM function by

$$Y(n) = X(n)\beta = X(n)\, X^T(\tfrac{1}{C}XX^T)^{-1}O \qquad (5)$$

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 10s
DOI: https://doi.org/10.17762/ijritcc.v11i10s.7695
***Article Received:** 08 June 2023 **Revised:** 29 July 2023 **Accepted:** 14 August 2023*

The general ELM training is delivered by
$S = \alpha(\sum_{n=1}^{N}(Y(n), B(n), W(n))$ (6)

To get the best accuracy, the feedforward layers are then combined with SoftMax activation layers.

SECTION -IV

IV. RESULTS AND DISCUSSION:

This section describes the proposed model's performance and extensive experimentation methods. A comprehensive comparative analysis of the different algorithms is also presented in this section.

4.2 EXPERIMENTATION:

The entire experiment is run on a PC workstation with a 3.2 GHz clock speed, an Intel I9CPU, a 256 GB NVIDIA Titan GPU, and 16 GB of RAM. This arrangement serves as the baseline station to test and validate the proposed model. Google Co-lab was used to train the suggested model. The various library packages, including TensorFlow 2.10, Keras 5.88, and OpenCV 1.10, are used. The proposed model is created and put into use using Caps net.

For training our suggested model, the ADAM optimizer is used as an optimizer. Table 3 displays the inputs that were utilized to train the recommended model. A number of metrics, including accuracy, precision, recall, specificity, and the F1-score, were calculated to optimize the performance of the proposed model. Table 4 displays the mathematical equations that were utilized to calculate the model's performance metrics.

Table 3 Training Hyper-Parameters used for Proposed Model

Sl.no	Hyperparameters	Values
1	Batch Sizes	30
2	No of Epochs	120
3	Learning Rate	0.0001
4	Loss Function Employed	Cross-Entropy
5	Momentum for ADAM optimizer	0.01
6	Drop-out	0.1

Table 4 lists the mathematical formulations for the performance metrics that are utilized for evaluation.

SL.NO	Performance Metrics	Mathematical Expression
01	Accuracy	$\frac{TP+TN}{TP+TN+FP+FN}$
02	Recall	$\frac{\mathrm{TP}}{\mathrm{TP+FN}}$ x100
03	Specificity	$\frac{TN}{TN+FP}$
04	Precision	$\frac{TN}{TP+FP}$
05	F1-Score	$2.\frac{Precison*Recall}{Precision+Recall}$

False Positive, True Negative, False Positive, and False Negative values are abbreviated as TP, TN, FP, and FN, respectively.

Nearly 70% of the data is utilized for training, 20% is used for validation, and 10% is used to assess the suggested mode. Although trained models at 50, 100, 150, and 200 iterations were also taken into consideration for comparison, the suggested model is trained for 250 iterations. A 100-epoch early termination method was used to avoid the over-fitting problem.

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 10s
DOI: https://doi.org/10.17762/ijritcc.v11i10s.7695
***Article Received:** 08 June 2023 **Revised:** 29 July 2023 **Accepted:** 14 August 2023*

4.3 PERFORMANCE EVALUATION:

Table 5 Performance Measures for the Suggested Model for Apple Plant Disease Detection

Plant and Disease type	**Measures of Performance**				
	Accuracy	**Precision**	**Recall**	**Specificity**	**F1-Score**
Healthy Apple	0.988	0.985	0.988	0.9882	0.9892
Apple Scab	0.988	0.984	0.987	0.9881	0.9892
Black rot	0.988	0.985	0.988	0.9882	0.9892
Ceder Apple xrust	0.988	0.984	0.987	0.9881	0.9892

Table 6 Performance Measures for the Suggested Model for Identifying Diseases in Strawberry Plants

Plant and Disease type	**Measures of Performance**				
	Accuracy	Precision	Recall	Specificity	F1-Score
Healthy Apple	0.988	0.985	0.988	0.9882	0.9892
Unhealthy Scab	0.988	0.984	0.987	0.9881	0.9892

Table 7 Performance Metrics for the Proposed Model for Detecting the Diseases in Corn Plant

Plant and Disease type	**Measures of Performance**				
	Accuracy	Precision	Recall	Specificity	F1-Score
Healthy Apple	0.988	0.985	0.988	0.9882	0.9892
Unhealthy Scab	0.988	0.984	0.987	0.9881	0.9892

Table 8 Performance Metrics for the Proposed Model for Detecting the Diseases in Squash Plant

Plant and Disease type	**Measures of Performance**				
	Accuracy	Precision	Recall	Specificity	F1-Score
Healthy Apple	0.988	0.985	0.988	0.9882	0.9892
Unhealthy Scab	0.988	0.984	0.987	0.9881	0.9892

Table 9 Performance Metrics for the Proposed Model for Detecting the Diseases in Squash Plant

Plant and Disease type	**Measures of Performance**				
	Accuracy	Precision	Recall	Specificity	F1-Score
Healthy Apple	0.988	0.985	0.988	0.9882	0.9892
Unhealthy Scab	0.988	0.984	0.987	0.9881	0.9892

Table 10 Performance Metrics for the Proposed Model for Detecting the Diseases in Soyabean Plant

Plant and Disease type	**Measures of Performance**				
	Accuracy	Precision	Recall	Specificity	F1-Score
Healthy Apple	0.988	0.985	0.988	0.9882	0.9892
Apple Scab	0.988	0.984	0.987	0.9881	0.9892

Table 11 Performance Metrics for the Proposed Model for Detecting the Diseases in Squash Plant

Plant and Disease type	**Measures of Performance**				
	Accuracy	Precision	Recall	Specificity	F1-Score
Healthy Potato	0.988	0.985	0.988	0.9882	0.9892
Light Blight	0.988	0.984	0.987	0.9881	0.9892
Early Blight	0.988	0.985	0.988	0.9882	0.9892

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 10s
DOI: https://doi.org/10.17762/ijritcc.v11i10s.7695
***Article Received:** 08 June 2023 **Revised:** 29 July 2023 **Accepted:** 14 August 2023*

Table 12 Performance Metrics for the Proposed Model for Detecting the Diseases in Potato Plant

Plant and Disease type	**Measures of Performance**				
	Accuracy	Precision	Recall	Specificity	F1-Score
Healthy Potato	0.988	0.985	0.988	0.9882	0.9892
Light Blight	0.988	0.984	0.987	0.9881	0.9892
Early Blight	0.988	0.985	0.988	0.9882	0.9892

Table 13 Performance Metrics for the Proposed Model for Detecting the Diseases in Tomato Plant

Plant and Disease type	**Measures of Performance**				
	Accuracy	Precision	Recall	Specificity	F1-Score
Healthy Tomato	0.988	0.985	0.988	0.9882	0.9892
Light Blight	0.988	0.984	0.987	0.9881	0.9891
Early Blight	0.988	0.985	0.988	0.9882	0.9891
Bacterial Spot	0.988	0.984	0.9889	0.9884	0.9890
Leaf Spot	0.9879	0.984	0.9882	0.9884	0.9889
Target Spot	0.9879	0.984	0.9886	0.9884	0.9892
Yellow Leaf Virus	0.988	0.985	0.9888	0.9882	0.9893
Mosaic Virus	0.988	0.984	0.9878	0.98832	0.9893
Spider Mates	0.988	0.985	0.9889	0.9882	0.9892

Tables 5 and 13 show how efficient the suggested technique is against various plant diseases. The suggested method, as shown in all the tables, has a high f1 score of 98.888% and has shown to be 98.8% accurate at differentiating between healthy and unhealthy images. Tables 5 and 13 clearly show that combining transfer learning with a capsule architecture has increased the detection ratio while beating other existing methods.

A COMPARISON ANALYSIS

Performance of the suggested model is compared to that of existing deep transfer learning and capsule networks to demonstrate its superiority. Examples of these networks include ResNets-50 [34], ResNet-100 [35], GoogleNets [36], AlexNets [37], Inception model [38], VGG-19 [39], CapsuleNetworks [40], CAPSNETS [41].

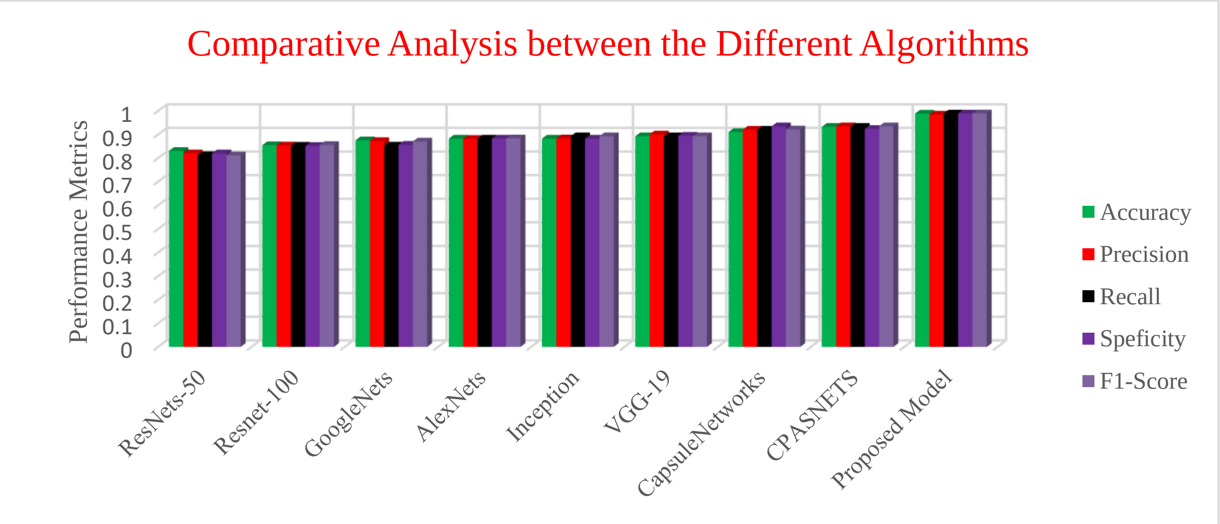

Figure 3 shows a comparison of various algorithms for detecting plant diseases in healthy plants

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 10s
DOI: https://doi.org/10.17762/ijritcc.v11i10s.7695
***Article Received:** 08 June 2023 **Revised:** 29 July 2023 **Accepted:** 14 August 2023*

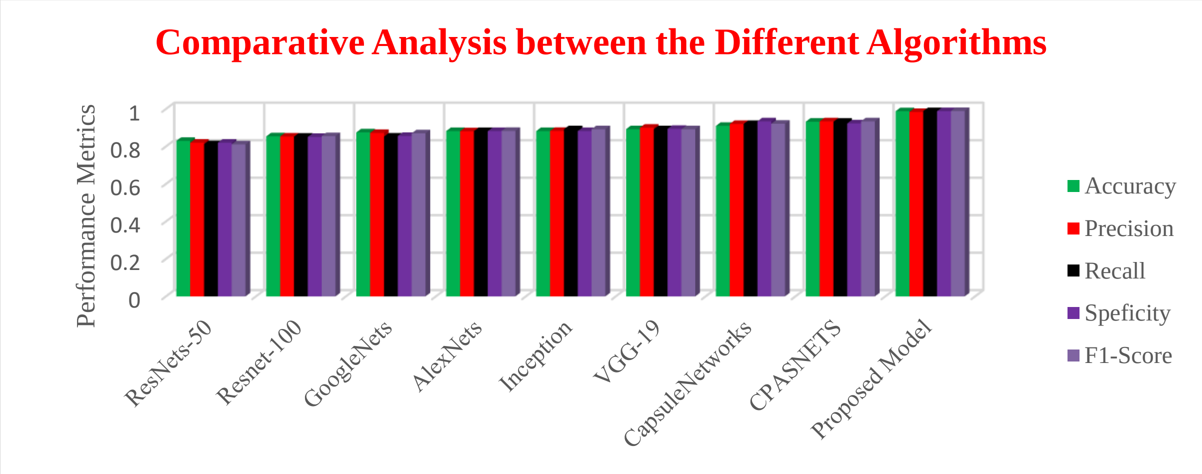

Figure 4 shows a comparison of various plant disease detection algorithms.

The average performance of the proposed and current algorithms is compared in Figures 3 and 4. It is also obvious from Figures 3 and 4 that the suggested approach has outperformed the previous algorithms.

SECTION -V

V. CONCLUSION AND FUTURE ENHANCEMENT:

Finding and categorizing healthy and diseased plant cells, which has an effect on agriculture, is the research's main goal. To achieve higher performance and a system free of complexity, this study recommends a novel combination of transfer learning with a capsule network and extreme feedforward networks. A plantation database is used to assess the performance of the suggested networks. The Keras API and TensorFlow 1.8 were used to develop the suggested algorithm, which was then evaluated against other cutting-edge designs. The findings demonstrate that the suggested architecture beat previous state-of-the-art designs and achieved the highest levels of performance, including 98.89% accuracy, 98.8% sensitivity and specificity, 98.86% precision, and 98.89% F1-score. The greater real-time clinical datasets will call for more rigorous testing in the future. Additionally, the suggested algorithm requires modification in order to handle even bigger real-time datasets and the real-time challenge.

Authors Contribution:

Each author contributed equally in each part.

Ethical Approval:

This article does not contain any studies with human participant and Animals performed by author.

Conflicts of Interest

The authors are declared there are no conflicts of interest regarding the publication of this paper.

Funding Statement

Author declared that no funding was received for this Research and Publication.

Acknowledgment

The authors thank for providing characterization supports to complete this research work.

REFERENCES:

[1] Mohanty SP, Hughes DP, Salathé M. 2016. Using deep learning for image-based plant disease detection. Frontiers in Plant Science 7:1419 DOI 10.3389/fpls.2016.01419.

[2] M. Ranjan et al., Detection and classification of leaf disease using artificial neural network, Int. J. Techn. Res. Appl. 3 (3) (2015) 331–333.

[3] B.S. Prajapati, V.K. Dabhi, H.B. Prajapati, A survey on detection and classification of cotton leaf diseases, 2016 International Conference on Electrical, Electronics, and Optimization Techniques (ICEEOT), IEEE, 2016.

[4] Ramya R. S., Parveen, M. S. ., Hiremath, S. ., Pugalia, I. ., S. H. Manjula, & Venugopal K. R. (2023). A Survey on Automatic Text Summarization and its Techniques . International Journal of Intelligent Systems and Applications in Engineering, 11(1s), 63–71. Retrieved from https://ijisae.org/index.php/IJISAE/article/view/2478

[5] J. Deng, W. Dong, R. Socher, L. Li, Kai Li and Li Fei-Fei, "ImageNet: A large-scale hierarchical image database," 2009 IEEE Conference on Computer Vision and Pattern Recognition, Miami, FL, 2009, pp. 248-255, doi: 10.1109/CVPR.2009.5206848.

[6] Ashish Vaswani and Noam Shazeer and Niki Parmar and Jakob Uszkoreit and Llion Jones and Aidan N. Gomez and Lukasz Kaiser and Illia Polosukhin, "Attention Is All You Need", 2017,https://arxiv.org/abs/1706.03762

[7] Bichen Wu and Chenfeng Xu and Xiaoliang Dai and Alvin Wan and Peizhao Zhang and Zhicheng Yan and Masayoshi Tomizuka and Joseph Gonzalez and Kurt Keutzer and Peter Vajda,"Visual Transformers: Tokenbased Image Representation and Processing for Computer Vision",2020
[8] Flores, A., Silva, A., López, L., Rodriguez, A., & María, K. Machine Learning-Enabled Early Warning Systems for Engineering Student Retention. Kuwait Journal of Machine Learning, 1(1). Retrieved from http://kuwaitjournals.com/index.php/kjml/article/view/106
[9] Konstantinos P. Ferentinos, "Deep learning models for plant disease detection and diagnosis",Computers and Electronics in Agriculture,Volume 145, 2018,Pages 311-318, ISSN 0168-1699,
[10] O. Carisse, T. Jobin, G. Bourgeois, Predicting apple leaf emergence from degreeday accumulation during the primary scab period, Canad. J. Plant Sci. 88 (1) (2008) 229–238.
[11] Vagisha Sharma, Amandeep Verma, Neelam Goel, Classification techniques for plant disease detection, Int. J. Recent Technol. Eng. (IJRTE). 8(6) (March 2020), ISSN: 2277-3878.
[12] M. Sardogan, A. Tuncer, Y. Ozen, Plant leaf disease detection and classification based on CNN with LVQ algorithm, 2018 3rd International Conference on Computer Science and Engineering (UBMK), IEEE, 2018.
[13] N. Satya Priya, E. Nivetha, R. Khilar, Efficient knowledge based system to detect diseases in lemon leaf, Imperial J. Interdiscip. Res. (IJIR) 2 (5) (2016)
[14] Y. Sun, Z. Jiang, L. Zhang, W. Dong and Y. Rao, "Slic_svm based leaf diseases saliency map extraction of tea plant," Computers and Electronics in Agriculture, vol. 157, pp.102-109, 2019, https://doi.org/10.1016/j.compag.2018.12.042.
[15] S. Ramesh, R. Hebbar, M. Nivetha, R. Pooja, N. P. Bhat et al., "Plant disease detection using machine learning," International Conference on Design Innovations for 3Cs Compute Communicate Control, Bangalore, India, pp. 41-45, 2018, doi: 10.1109/ICDI3C.2018.00017.
[16] G. Kuricheti and P. Supriya, "Computer vision based turmeric leaf disease detection and classification: a step to smart agriculture," International Conference on Trends in Electronics and Informatics, Tirunelveli, India, pp. 545-549, 2019, doi: 10.1109/ICOEI.2019.8862706.
[17] Prof. Naveen Jain. (2013). FPGA Implementation of Hardware Architecture for H264/AV Codec Standards. International Journal of New Practices in Management and Engineering, 2(01), 01 - 07. Retrieved from http://ijnpme.org/index.php/IJNPME/article/view/11
[18] D. Argueso, A. Picon, U. Irusta, A. Medala, M. G. S. Emeterio et al., "Few-shot learning approach for plant disease classification using images taken in the field," Computers and Electronics in Agriculture, vol. 145, article ID. 105542, pp.1-8, 2020.
[19] Hiroshi Yamamoto, An Ensemble Learning Approach for Credit Risk Assessment in Banking , Machine Learning Applications Conference Proceedings, Vol 1 2021.
[20] Vyas, A. ., & Sharma, D. A. . (2020). Deep Learning-Based Mango Leaf Detection by Pre-Processing and Segmentation Techniques. Research Journal of Computer Systems and Engineering, 1(1), 11–16. Retrieved from https://technicaljournals.org/RJCSE/index.php/journal/article/view/18
[21] B. Richey, S. Majumder, M. Shriviakar and N. Kehtarnavaz, "Real-time detection of maize crop disease via a deep learning-based smartphone app," Proceedings of SPIE, vol.11401, pp. 114010A-1-114010A-7, 2020.
[22] Y. Zhang, C. Song and D. Zhang, "Deep learning-based object detection improvement for tomato disease," IEEE Access, vol. 8, pp. 56607-56614, 2020, doi: 10.1109/ACCESS.2020.2982456.
[23] A. Batool, S. B. Hyder, A. Rahim, N. Waheed, M. A. Asghar et al., "Classification and identification of tomato leaf disease using deep neural network," International Conference on Engineering and Emerging Technologies, Lahore, Pakistan, pp. 1-6, 2020, doi: 10.1109/ICEET48479.2020.9048207.
[24] R. Karthik, M. Hariharan, S. Anand, P. Mathikshara, A. Johnson et al., "Attention embedded residual cnn for disease detection in tomato leaves," Applied Soft Computing, vol. 86, article ID. 105933, pp.1-27, 2020. doi: https://doi.org/10.1016/j.asoc.2019.105933.
[25] M. Turkoglu, B. A. Yanikoglu and D. Hanbay, "Plantdiseasenet: convolutional neural network ensemble for plant disease and pest detection," Signal, Image and Video Processing, vol.16, no.9, pp. 1-9, 2022.
[26] S. Verma, A. Chug and A. P. Singh, "Exploring capsule networks for disease classification in plants," Journal of Statistics and Management Systems, vol.23, no.2, pp. 307 – 315, 2020, DOI: 10.1080/09720510.2020.1724628.
[27] L. W. Waweru, B. T. Kipyego and D. M. Muchangi, "Classification of plant leaf diseases based on capsule network-support vector machine model," International Journal of Electrical Engineering and Technology, vol. 12, no.6, pp.188-199, 2021.
[28] G. Altan, "Performance evaluation of capsule networks for classification of plant leaf diseases", International Journal of Applied Mathematics, Electronics and Computers, vol.8, no.3, pp. 057-063, 2020.
[29] P. M. Kwabena and B. A. Weyori, "Gabor capsule network for plant disease detection," International Journal of Advanced Computer Science and Applications, vol. 11, no. 10, pp. 388-395, 2020
[30] S. Sabour, N. Frosst and G. E. Hinton, "Dynamic routing between capsules," in Proceedings of Conference on Neural Information Processing Systems, Long Beach, CA, USA, pp. 1- 11, 2017. L.
[31] L. Jovanovic, N. Bacanin, M. Antonijevic, E. Tuba, M. Ivanovic and K. Venkatachalam, "Plant Classification Using Firefly Algorithm and Support Vector Machine," 2022 IEEE Zooming Innovation in Consumer Technologies Conference (ZINC), 2022, pp. 255-260, doi: 10.1109/ZINC55034.2022.9840579.
[32] G. B. Huang, Q.-Y. Zhu and C.-K. Siew, "Extreme learning machine: theory and applications," Neurocomputing, vol. 70, no. 1, pp. 489–501, 2006.

[33] B. Wang, S. Huang, J. Qiu, J. Yao, G. Wang and G. Yu, “Parallel online sequential extreme learning machine based on MapReduce,” Neurocomputing, vol. 149, pp. 224-32, 2015.
[34] S. M. Hassan and A. K. Maji, "Plant Disease Identification Using a Novel Convolutional Neural Network," in IEEE Access, vol. 10, pp. 5390-5401, 2022.
[35] H. Yu et al., "Corn Leaf Diseases Diagnosis Based on K-Means Clustering and Deep Learning," in IEEE Access, vol. 9, pp. 143824-143835, 2021, doi: 10.1109/ACCESS.2021.3120379.