Journal of Autonomous Intelligence (2024) Volume 7 Issue 4
doi: 10.32629/jai.v7i4.1126

ORIGINAL RESEARCH ARTICLE

Single and multi-crop species disease detection using ITSO based gated recurrent multi-attention neural network

B. Rajalakshmi[1], Santosh Kumar B.[1], B. S. Kiruthika Devi[2,*], Balasubramanian Prabhu Kavin[3], Gan Hong Seng[4]

[1] *New Horizon College of Engineering, Bengaluru 560103, India*

[2] *School of Computing, Sathyabama Institute of Science and Technology, Chennai 600119, India*

[3] *Department of Data Science and Business Systems, SRM Institute of Science and Technology, Kattankulathur, Chengalpattu, Tamilnadu 603203, India*

[4] *School of AI and Advanced Computing, XJTLU Entrepreneur College (Taicang), Xi'an Jiaotong-Liverpool University, Suzhou 215400, Jiangsu, China*

*** Corresponding author:** B. S. Kiruthika Devi, kiruthikadevi.b.s.cse@sathyabama.ac.in

ABSTRACT

Diseases of crop plants pose a serious danger to agricultural output and progress. Predicting the onset of a disease outbreak in advance can help public health officials better manage the pandemic. Precision agriculture (PA) applications rely heavily on current information and communication technologies (ICTs) for their contribution to long-term sustainability. Preventative measures against plant diseases require accurate early disease prediction in order to be effective. The current computer vision-based illness detection technology can only detect the disease after it has already manifested. This research intends to provide a deep learning (DL) method for early disease attack prediction using Internet of Things (IoT) directly sensed environmental factors from crop fields. There is a robust relationship between environmental factors and the life cycles of plant diseases. Disease incidence in plants can be forecast based on environmental variables in the crop field. In order to solve these issues, the research presented here suggests using a gated recurrent multi-attention neural network (GRMA-Net). The study uses multilevel modules to zero down on informative areas in order to extract additional discriminative features, as informative characteristics tend to appear at various levels in a network. In order to capture long-range dependence and contextual interaction, these characteristics are first organised as spatial sequences and then input into a deep-gated recurrent unit (GRU). Finally, an enhanced version of the Tunicate swarm optimisation model (ITSO) is used to pick the best values for the model's hyper-parameters. Four public datasets representing a wide range of crop types are used to assess the model's efficacy. Some of these databases cover numerous crop species, like PlantVillage (38 categories), while others focus on a single crop, such as Apple (4), Maize (4), or Rice (5). The experimental findings show that the system achieves 99.16% accuracy in identifying agricultural diseases, which is higher than the accuracy of other current deep-learning approaches.

Keywords: information and communication technologies; precision agriculture; Internet of Things; multi-attention neural network; deep-gated recurrent unit; improved tunicate swarm optimization

1. Introduction

In a lot of places, agriculture is the economy. The rising number of people need more nourishment. The only way to meet this urgent demand is to boost agricultural output and implement crop protection measures[1]. However, a wide variety of pathogens in crops' natural habitats makes them vulnerable to a wide range of illnesses. Disease-causing microorganisms can take several forms, including viruses, fungi, and bacteria[2]. There is a direct correlation between the prevalence of crop diseases and a precipitous drop in agricultural output, with losses ranging from 10% to 95%[2]. In order to prevent massive losses and cut down on the overuse of potentially dangerous pesticides, early disease detection is essential. Predicting the onset of disease outbreaks in order to implement preventative measures, reduce the illness's impact, and promote long-term

ARTICLE INFO

Received: 9 August 2023
Accepted: 11 December 2023
Available online: 29 January 2024

growth is crucial[3]. Most of the time, and especially in less developed nations and on smaller farms, farmers will use just their eyes to diagnose crop illnesses. This is a time-consuming process that calls for plant pathology knowledge and extensive treatment time[4]. Furthermore, if a rare disease is attacking the field, farmers will seek specialist help to get a precise and quick diagnosis, which will inevitably lead to higher treatment expenses. Therefore, huge farms cannot reasonably use this type of observation, and it may even lead to inaccurate forecasts as a result of subjective judgements[5]. To meet rising consumer demands and lessen the environmental impact of chemical inputs while protecting human health, researchers have developed technological proposals for early identification of crop diseases in an accurate, fast, and reliable manner[6]. Direct visual diagnosis, in which disease signs on leaves are seen visually, and chemical approaches, including molecular analyses on leaves, are two common ways of identifying and localising plant diseases. Time-consuming and labour-intensive[7] describes these approaches.

Using autonomous monitoring and recognition systems, promising methods for identifying and localising illnesses have been developed in recent years. Recent developments in sensor and data processing have provided new opportunities for the early identification and accurate diagnosis of agricultural abnormalities[8]. Developing and testing machine learning algorithms[9] and collecting data from sensors like remote sensing (RS) or ground equipment are all viable options for disease monitoring. Profitability, sustainability, and conservation of land resources may all be improved by the use of smart algorithms in management practices. Together, these factors make it possible to administer curative care at optimal times and locations[10]. Multiple ambient, plant canopy, and leaf indices derived from remote sensing photography, as well as Internet of Things (IoT) sensors, may be given to the agriculture field. To better understand crop growth circumstances and disease symptom development, a range of data extraction methodologies must be combined using data fusion techniques[11]. The use of machine learning-based data fusion to agricultural data has the potential to significantly improve plant protection, especially in the areas of disease and early disease detection[12]. For this reason, several fusion approaches based on distant sensing and many sensors have been used in agriculture[13].

Researchers have proposed a number of methods, many of which make use of image processing and machine learning (ML). The availability of data, processing power, and sophistication in the learnability of ML algorithms have all contributed to the remarkable success of recent techniques[14]. In the field of agricultural disease diagnosis, many methods have stood out: support vector machines (SVM), K-nearest neighbours (KNN), multilayer perceptrons (MLPs), fisher linear discriminants (FLDs), and random forest classifiers[15].

In recent years, deep learning (DL) has been a popular solution to many computer vision issues. Similarly, deep convolutional neural networks (CNNs) have demonstrated encouraging results in crop disease identification[16]. Over the past several years, many new techniques for illness diagnosis using deep learning have been developed[17]. These first researches and subsequent efforts sparked a growing interest in using deep learning frameworks for the detection and diagnosis of agricultural diseases. The geographical link and contextual reliance of these characteristics are underutilised by current approaches[18]. Indeed, these widely dispersed regions typically exhibit robust spatial linkage and contextual reliance, both of which are necessary for precise categorization.

We create a multilayer attention module to interest across several dimensions to solve the first issue. Local characteristics can be directed towards informative signals by using the high-level semantic material derived by global features[19]. The significant magnitude disparity among multiscale would decrease the guiding of global features if we directly integrate them to build attention map. As a result, in order to fine-tune local features during feature aggregation, we offer an adaptive convolution. The study uses a recurrent neural network (RNN) to take advantage of the connection between distinct sites, inspired by the success of RNNs in simulating long-range dependence[20]. We reformat multi-scale characteristics into spatial sequences and then apply a deep RNN to each sequence individually.

In conclusion, the article's contribution may be summed up as follows:

(1) Weak reliance among broadly distributed informative features is addressed by the study's proposed gated recurrent multi-attention neural network (GRMA-Net).
(2) To assign discriminative leverage, the spatial dependence of features at various modules are presented.
(3) ITSOA model is used to fine-tune the proposed model's hyper-parameters by introducing a Cauchy mutation operator to address the issue of global convergence.

Here is how the rest of the paper is structured: In section 2, we provide the literature that has addressed this issue, and in section 3, we provide a brief summary of the suggested model. The results of the experiments and their discussion may be found in section 4. In section 5, we provide the results of our investigation.

2. Related works

For the purpose of identifying agricultural illnesses from pictures of plant leaves, the lightweight convolutional neural network 'VGG-ICNN' is introduced by Thakur et al.[21]. Compared to other high-performing deep learning models, VGG-ICNN has a lot less parameters (about 6 million) to work with. Five public datasets representing a wide range of crop types are used to assess the model's efficacy. PlantVillage and Embrapa, which cover various crop species, have 38 and 93 categories, respectively, while Apple, Maize, and Rice, which focus on a single crop, have 4, 4, and 5 categories, respectively. With an accuracy of 99.16% on the PlantVillage dataset, experimental findings show that the system exceeds some of the latest deep-learning algorithms on crop disease diagnosis. Comparing the model's results to those of more contemporary lightweight CNN models demonstrates that it outperforms them consistently across all five datasets.

A unique two-stage semantic segmentation technique has been developed by Divyanth et al.[22] to detect maize illnesses and quantify their impact. SegNet, UNet, and DeepLabV3+ network architectures were used to train the three semantic segmentation models used in each phase. The first stage was the use of semantic segmentation to remove leaves from their murky field settings. Second, disease lesions were found, identified, and their area coverage was determined using semantic segmentation. Following model training, the UNet model showed the best stage 1 performance, with a mean of 0.9422 and a mean boundary F1-score (mBFScore) of 0.8063. The DeepLabV3+ model performed the best in stage two, with a mwIoU of 0.7379 and a mBFScore of 0.5351 for detecting disease lesions. Finally, the proportion of leaf area that was affected by disease lesions was used to evaluate severity. The combined (UNet-DeepLabV3+) model predicted the severity of three

illnesses fairly near to the actual data, as indicated by an R2 value of 0.96 in the test set. An innovative two-stage deep learning-based technique was created in this work to properly identify three specific maize illnesses and quantify their severity. This research paves the way for the creation of a disease management system suitable for use in the field.

WheatRust21 is a dataset created by Nigam et al.[23] that contains 6556 photos of both healthy and damaged leaves collected in field settings. They also tested many traditional CNN-based models for identifying Wheat rust, achieving accuracy between 91.2% and 97.8% using VGG19, ResNet152, DenseNet169, InceptionNetV3, and MobileNetV2. They tried out eight different variations of the EfficientNet architecture in an effort to boost accuracy and found that our optimised EfficientNet B4 model was able to attain a testing accuracy of 99.35%, a value that has not been published in the literature so far. Stakeholders may utilise this model for image-based wheat disease diagnosis in field situations by integrating it into mobile applications.

To improve crop disease identification, Zhao et al.[24] introduced a YOLOv5s-based model. First, during the feature fusion step of the original model, we used a lightweight structure in the improved CSP structure to reduce the model parameters while extracting the feature information of different layers in the form of multiple branches. A new structure called CAM was developed to aid in the extraction of global information from networks. CAM is able to better combine semantic and scale-inconsistent information, as well as extract global and local properties from separate network layers. The original model, which only used three grids to predict the target, was expanded to include a fourth grid, and the formula for the prediction frame centroid offset was modified to obtain a better offset when the target centroid was located on the grid's special point. To fix the problem of the prediction frame being scaled incorrectly, we upgraded the YOLOv5s model training process by using the DIoU loss function instead of the GIoU loss function. The improved model beat the Faster R-CNN, SSD, YOLOv3, YOLOv4, YOLOv4-tiny, and YOLOv5s models with mAP values of 95.92%, F1 score values of 0.91, and recall values of 87.89% after being trained using transfer learning. When it comes to diagnosing diseases that might affect tomato leaf tissue, Sanida et al.[25] provides a powerful hybrid convolutional neural network (CNN) tool, with findings that are superior to YOLOv5s' by 4.58%, 5%, and 4.78%. A convolutional neural network (CNN) and an inception module have been combined to form this hybrid technique. The dataset used in this analysis comes from PlantVillage and includes nine disease categories and one healthy category for tomatoes. Results on the test set are encouraging, with an accuracy of 99.17%, recall of 99.23%, precision of 99.13%, area under the curve of 99.56%, and F1-score of 99.17%. The suggested approach provides a high-performance solution for diagnosing tomato crops in a real-world agricultural scenario.

An integrated deep learning technique (RFE-CNN) has been proposed by Xu et al.[26]; At first, we extracted the most fundamental properties of both healthy and damaged wheat leaves using two simultaneous convolutional neural networks. Second, we employed attention blocks in the residual channel to fine-tune the foundational characteristics. Third, we trained the earlier features with feedback blocks. To complete the processing and classification, we next fed these characteristics into a convolutional neural network and elliptic metric learning. The experimental findings show that the suggested model outperforms the industry standard VGG-19, ZFNet, GoogLeNet, Inception-V4, and Efficient-B7 in several respects, including reduced training time, improved accuracy during recognition, and enhanced flexibility. The maximum testing accuracy was 99.95%, while the total classification accuracy was 98.83 per cent. On the publicly available CGIAR, Plant Diseases, LWDCD 2020, and Plant Pathology datasets, we were able to achieve an average precision of 99.50 percent.

Using transfer learning, Zhang et al.[27] pre-trained the model using the PDDA and PlantVillage datasets, resulting in an enhanced PlantDoc++ dataset. Pre-training on the PDDA dataset led to IBSA_Net's 0.946 test accuracy on a dataset from the wild, with averages of 0.942 for precision, 0.944 for recall, and 0.943 for F1-score. The efficiency of IBSA_Net has also been demonstrated in other crop systems. This research establishes

a reliable and practical approach for diagnosing leaf diseases in commercially grown tomatoes; the findings may be applied to other crops as well.

The model presented by Joshi and Bhavsar[28] is intended to improve upon previous methods for identifying leaf diseases in plants. The Night shed plant's leaf was used to train the state-of-the-art AlexNet, VGG, and GoogleNet models, as well as the suggested model. There are 9 different groups of illnesses and normal leaf conditions. The success or failure of the models was judged using many criteria, such as the number of training samples, the number of training failures, the learning rate, and the activation function. Disease categorization accuracy of between 93% and 95% was attained by using the suggested approach. The results of the precision tests indicate that the proposed model has promise and has the potential to significantly impact the efficiency and precision with which diseased leaves are detected.

To perform crop disease identification against a complicated backdrop, Ma et al.[29] offer a neural network-based technique that incorporates an enhanced rouse pyramid pooling algorithm. To facilitate the extraction of multi-dimensional illness info from the channel and space perspectives, a dual-attention module was initially added to the cross-stage partial network backbone during the process of building neural networks. To further enhance the network's capacity to extract useful information about crop diseases from field photos, a dilated pyramid pooling module was subsequently incorporated into the network. The neural network was evaluated using a dataset made up of photos captured at 40 frames per second, taking up only 17.12 MB of space. When compared to results acquired using equivalent traditional approaches, the average accuracy rate revealed by field data analysis utilising the miniaturised model was nearer to 90.15 per cent. When taken as a whole, these findings suggest that the model simplifies disease-recognition tasks while suppressing noise broadcast to attain a higher accuracy rate than similar methods, suggesting that the projected method should be suitable for use in applied applications of crop disease acknowledgement.

To improve the disease diagnosis method for tomatoes, Sanida et al.[30] employ transfer learning to shorten model training time while simultaneously increasing identification accuracy. VGGNet, which has been pre-trained with data from ImageNet, and two inception blocks form the basis of the model. Model training also used two-stage transfer learning and an enhanced. The results of the studies demonstrate that our model achieves higher for the test set of tomato illnesses than other state-of-the-art methodologies. The results show that the proposed method yields significant values with an accuracy of 99.23%. Farmers may use the proposed model to their advantage in the fight against disease in tomatoes since it has a very high success rate.

Based on the meta-learning paradigm, Si et al.[31] present a Model-Agnostic Meta-Learning (MAML) attention model. The proposed model incorporates both meta-learning and conventional training, using an Efficient Channel Attention (ECA) component for the latter. To improve the weight parameters corresponding to specific illness features, the module employs a local cross-channel interaction technique of nondimensional reduction. By including the ECA module in the original model, the proposed meta-learning-based approach is able to accomplish more efficient detection in new tasks and benefit from its great generalisation capabilities. Experiments validate the proposed model, revealing that it outperforms the original MAML model by a margin of 1.8–9.31 percentage points across a variety of classification tasks, with an increase in maximum accuracy of 1.15–8.2 percentage points. The experimental outcomes validate the suggested MAMLAttention model's high generalisation ability and superior resilience. The suggested MAML-Attention approach outperforms the existing few-shot methods.

A technique for rice disease diagnosis using an enhanced DenseNet network (DenseNet) was proposed by Jiang et al.[32]. This technique takes DenseNet as its baseline model and employs the channel attention mechanism of squeeze-and-excitation to amplify the positive features while dampening the negative ones. The typical convolutions in the dense network are then replaced with depth-wise separable convolutions to boost parameter utilisation and training speed. The AdaBound algorithm, in conjunction with the adaptive

optimisation technique, expedites the tuning process. This paper's approach improves upon the previous model by 13.8% in terms of classification accuracy when tested on five different types of rice disease datasets. Simultaneously, it is evaluated next to established recognition strategies like ResNet, VGG, and Vision Transformer. This approach improves recognition accuracy, allows for efficient picture classification of rice diseases, and presents a novel strategy for advancing crop disease identification technology and smart agriculture.

Problem statement

Researchers and experts analyse plant leaf diseases to identify the most pressing problems and obstacles. Here are a few examples:

- One, the picture of the leaf has to be of very good quality.
- Availability of the dataset to the public is a must.
- The leaf samples had 3 noise in the data. Diseases may be recognised via segmentation, but only after samples have been trained and tested.
- In the process of identifying leaf diseases, the fifth hurdle is categorization. The environment can cause variation in leaf colour by six shades.

It might be challenging to identify all of the many plant diseases that exist.

The suggested model's improved precision is based on the aforementioned difficulties and on the fusion of IP and ML approaches. In this study, we consider these factors and present a DL-based system for autonomous leaf disease detection.

3. Proposed system

Here, you'll find a machine-learning and image-processing architecture for identifying leaf diseases. The input for this framework is a picture of a leaf. The first step is a noise reduction procedure applied to the leaf pictures. The mean filter[33] is used to get rid of the noise. Histogram equalisation[34] is a technique used to improve images. The process of picture segmentation involves cutting a single image into several smaller sections. It aids in pinpointing the limits of a picture. K-Means is a method for image segmentation[35]. The suggested deep learning model is then used to classify images.

The adaptive median filter (AMF) algorithms[33] are widely used to clean up pictures that have been tainted by background noise. The AMF technique uses this type of spatial processing to determine which pixels in an image are impacted by impulse noise. Impulse noise is the result of a large number of misaligned pixels in a given area. As a result, the median value of neighbouring, noise-free pixels are used to mask the noise-containing pixels.

Histogram equalisation, which is used to boost contrast, coordinates the intensity values of individual pixels to produce a uniform intensity profile and smooth histogram in the final picture[34]. When the contrast values of the picture's practical data are extremely high, this method is frequently employed to increase the image's contrast. This method has the potential to create a uniform distribution of brightness[35]. As a result, more contrast may be necessary in some parts of the image. Histogram equalisation is a method for more uniformly dispersing the most frequently occurring intensity values over the whole histogram.

3.1. Classification of plant disease

In this study, we create a multilayer attention component to help the network focus on the most important details while ignoring the rest. In addition, the research suggests a recurrent module that uses spatial relationships and contextual dependencies among informative portions of a pre-processed picture to improve accuracy. **Figure 1** depicts the general architecture of the suggested technique.

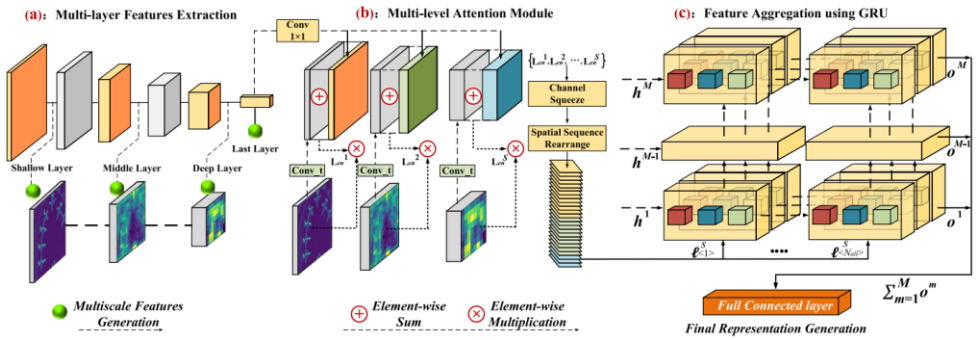


Figure 1. Illustration of the proposed gated recurrent multi-attention network.

3.1.1. Overall architecture

Multiscale local characteristics are extracted from input photos by feeding them into a core convolutional neural network. $L^s \in R^{C_s \times H_s \times W_s}$ and global feature $G \in R^{C_g \times 1 \times 1}$. Features $L^s \in R^{C_s \times H_s \times W_s}$ ($s \in$ {1, 2, 3, ..., S}) at single generate L^s 0. The global feature $G \in R^{C_g \times 1 \times 1}$ is fed into a 1 × 1 convolution to generate $G_0 \in R^{C_s \times 1 \times 1}$. And then is stretched to the size of $G_1 \in R^{C_s \times H_s \times W_s}$. After element-$L_0^s$ and G_1, the obtained score map F^s is multiplication $L_{en}^s = \alpha^s \otimes L^s$, the enhanced multiscale features $L_{en} = \{L_{en}^1, L_{en}^2, \ldots, L_{en}^S\}$ are obtained. Multiscale sequences $L_{en} = \{\ell_1, \ell_2, \ldots, \ell_{Nall}\}$. These sequences are then input into deep GRUs, which scour the environment for the most relevant contextual dependencies and spatial associations. The image label is gotten by $Y = GRU(L_{en})$.

3.1.2. Multiscale feature extraction

Multiple stacked layers make up the multiscale feature extraction component. As the network evolves, its focus shifts from local texturing to more global profiles. The study developed a multilevel attention module to enhance the multiscale illustration capabilities of backbone networks since these qualities are crucial to plant disease picture categorization. Our module's attention operation is fed into the system by first extracting multiscale local characteristics. Here, we have the expression for the scale *s*:

$$L^s = \{I_1^s, I_2^s, I_3^s, \ldots, I_{N_s}^s\} \tag{1}$$

where $C_s, H_s,$ and W_s denote the sum of channels of L^s, respectively. I_n^s represents the value of local feature L^s at spatial site $n \in \{1, 2, 3, \ldots, N_s\}$, at a given layer $s \in \{1, 2, 3, \ldots, S\}$. Then, feature $G \in R^{C_g \times 1 \times 1}$ is also produced by the first non-layer before the softmax layer. C_g represents the channels of *G*.

3.1.3. Multilevel attention module

Let's pretend that *L* stands for the coarse local feature and *G* for the fine global feature. Global features' extracted semantic high ground can help direct local features' attention to relevant indicators. Directly combining multiscale local characteristics with global features to produce an attention map is problematic due to the significant magnitude difference between the two sets of features.

Therefore, we first feed the local features $L_s \in R^{C_s \times H_s \times W_s}$ into Conv_t to their magnitudes at scale *s*, subsequent in L_0^s.

$$L_0^s = \text{Conv\_t}(L^s), \qquad L_0^s \in R^{C_s \times H_s \times W_s} \tag{2}$$

The global feature *G* is fed to a 1 × 1 convolution to produce $G_0 \in R^{C_s \times 1 \times 1}$. Then, G_0 is stretched to the size of $G_0 \in R^{C_s \times H_s \times W_s}$. After element-wise sum among L_0^s and G_1. The score map Fs at scale *s* can be generated by rendering to

$$F^s = \sigma(L_0^s + G_1) \tag{3}$$

where σ is the ReLU function. Once $F = \{F_1, F_2, \ldots, F_S\}$ is generated, a map

$$a_n^s = \frac{\exp(f_n^s)}{\sum_{n=1}^{N_s} \exp(f_n^s)}, n \in \{1,2,3,\ldots,N_s\} \quad (4)$$

where f_n^s signifies the score map F_n^s at location n, at a given scale $s \in \{1, 2, 3, \ldots, S\}$.

Finally, we do a multiplication of local features by their associated normalised attention weight value, *s n*. I_n^s. That is, $L_{en}^s = \{\ell_1, \ell_2, \ldots, \ell_{N_s}\}$ is produced as the final form for the image at each scale s.

3.1.4. Feature aggregation using GRU

We have successfully retrieved enough multiscale information from pictures over large geographic ranges to use in the multilevel attention module. It is a challenge to figure out how to more effectively merge these disparate characteristics. Informational dependencies are easy for RNN to capture. GRU is a subset of RNNs that, thanks to its memory for distant data, can outperform more conventional RNN architectures. We employ GRU in our network to progressively analyse these multiscale properties and automatically determine the ideal iteration, allowing us to fully capitalise on the long-range interdependence among pieces of material.

Features extracted by the multi-attention module may be seen as spatial series, analogous to how GRU is used in natural language processing to organise features in temporal series. Researchers started by using an 11-convolution procedure to compress multiscale feature channels. $L_{en} = \{L_{en}^1, L_{en}^2, \ldots, L_{en}^s\} \in R^{C_{en} \times H_{en} \times W_{en}}$ into a single channel and generated $L_{en} \in R^{1 \times H_{en} \times W_{en}}$. Then, the single $L_{en} = \{\ell_1, \ell_2, \ldots, \ell_{N_1}, \ell_1, \ell_2, \ldots, \ell_{N_s}, \ell_1, \ell_2, \ldots, \ell_{N_{all}}\} \in R^{1 \times (H_{en} W_{en})}$. For feature ℓn at recurrence can be formulated as

$$\tilde{h}_{\langle n,l \rangle}^m = \tanh\left(W_c\left[\Gamma_r \times h_{\langle n-1,l \rangle}^m, \ell_{\langle n,l \rangle}^m\right] + b_c\right) \quad (5)$$

$$\Gamma_u = \sigma\left(W_u\left[h_{\langle n-1,l \rangle}^m, \ell_{\langle n,l \rangle}^m\right] + b_u\right) \quad (6)$$

$$\Gamma_r = \sigma\left(W_u\left[h_{\langle n-1,l \rangle}^m, \ell_{\langle n,l \rangle}^m\right] + b_r\right) \quad (7)$$

$$h_{\langle n-1,l \rangle}^m = \Gamma_u \times \tilde{h}_{\langle n,l \rangle}^m + (1 - \Gamma_u) \times h_{\langle n-1,l \rangle}^m \quad (8)$$

$$O_{\langle n,l \rangle}^m = \text{sigmoid}\left(W_o \times h_{\langle n,l \rangle}^m + b_c\right) \quad (9)$$

Note that $h_{\langle n,l \rangle}^m$, $\ell_{\langle n,l \rangle}^m$, $O_{\langle n,\text{and } l \rangle}^m$ represent the *m*-th recurrence, the *n*-th layer, and the *n*-th spatial locations respectively. The update gate is denoted by *_u* and the reset gate by *_r*. These settings decide whether the hidden state *h_n*, *lm* should be remembered or forgotten at each spatial step.

Then, the hidden state $h_{\langle n,l \rangle}^m$ to produce the hidden state hm of the final layer and the output om of the *m*-th repetition iteration.

$$h^m = \{h_{[N_{all},1]}^m, h_{[N_{all},2]}^m, \ldots, h_{[N_{all},L]}^m\} \quad (10)$$

$$o^m = \{o_{[1,L]}^m, o_{[2,L]}^m, \ldots, o_{[N_{all},L]}^m\} \quad (11)$$

Hence, at the (*m* + 1)-th recurrence, the hidden state *hm* of the previous layer is considered the initial hidden state. The output o^M after *M* iterations looks like this

$$o^M = \left\{o_{\langle 1,L \rangle}^M, o_{\langle 2,L \rangle}^M, \ldots, o_{\langle N_{all},L \rangle}^M\right\} \quad (12)$$

Finally, o^M the output is the result of summing the over *M* iterations and running the result via a fully linked layer.

$$Y = FC\left(\sum_{m=1}^{M} o^M\right) \quad (13)$$

The parameters of the proposed model are optimally selected by the improved tunicate swarm optimization, which is explained as follows:

3.1.5. Tunicate swarm algorithm

To model the marine existence of the tunicate and its foraging behaviour, the tunicate swarm algorithm (TSA) was developed[36]. In the ocean, tunicates may look for food even if they have no prior knowledge of where to get it. The TSA algorithm employs two jet propulsion and swarm intelligence behaviours seen in tunicates to perform optimisation tasks. Tunicate must follow three major requirements in order to mathematically algorithm jet pulsing behaviour: travelling towards the position of the best search agent, among search agents, and residual near to the best search agent. During congestion behaviour, the optimal placement of additional search agents is determined. The underlying mathematical method behind these behaviours is detailed below. Vector *A* is employed in Equation (14) to determine the new location of the search agent so that it does not collide with other search agents (other tunicates).

$$\vec{A} = \frac{\vec{G}}{\vec{M}} \tag{14}$$

In Equation (1), the vector $\vec{G}$, which is gravity, is gotten based on Equation (15).

$$\vec{G} = c_2 + c_3 - \vec{F} \tag{15}$$

In Equation (15), the vector $\vec{F}$ the water depth is gotten based on Equation (16).

$$\vec{F} = 2c_1 \tag{16}$$

In Equations 15 and 16, the variables c_1, c_2, and c_3 are random values betwixt 0 and 1. Lastly, the vector $\vec{M}$ signifies the social force among the search agents, which computes using Equation (17).

$$\vec{M} = [P_{\min} + c_1 P_{\max} - P_{\min}] \tag{17}$$

In Equation (17), $P_{\max}$ and $P_{\min}$ characterise the primary and secondary velocities to generate social interaction. In the TSA procedure, $P_{\max}$ and $P_{\min}$ values reflect 1 and 4 correspondingly. After the search agent's novel position, compute the search agent's move to the best neighbour exposed in Equation (18).

$$\overrightarrow{PD} = \left|\overrightarrow{FS} - \text{rand}(). \vec{P}_p(x)\right| \tag{18}$$

In Equation (18), $\overrightarrow{PD}$ shows how far away the meal is from the person doing the searching. $P_p(x)$ represents the tunicate's position relative to the ideal location of the food supply, where *x* is the current repeat value. The TSA algorithm then moves on to the next step, in which the search agents converge on the agent so that it can retain its position with respect to the best search agent—the food source. The algorithm at this point operates in the context of Equation (19).

$$\vec{P}_p(x) = \begin{cases} \overrightarrow{FS} + \vec{A}. \overrightarrow{PD}, & \text{if } r_{\text{and}} \geq 0.5 \\ \overrightarrow{FS} - \vec{A}. \overrightarrow{PD}, & \text{if } r_{\text{and}} \leq 0.5 \end{cases} \tag{19}$$

In Equation (19), $\vec{P}_p(x)$ is the efficient position $\overrightarrow{FS}$. Finally, two of the most promising optimum solutions are kept in a database throughout the mathematical modelling phase of swarm behaviour, and the locations of the other search agents are adjusted accordingly. Algorithmically, this behaviour is represented by Equation (20).

$$P_p(\vec{x} + 1) = \frac{\vec{P}_p(x) + P_p(\vec{x} + 1)}{2 + c_1} \tag{20}$$

3.1.6. Cauchy mutation-TSA (QCTSA)

In this part, the CTSA algorithm undergoes the Cauchy mutation. Equation (21) demonstrates the Cauchy mutation.

$$y = \frac{1}{2} + \frac{1}{\pi} \arctan\left(\frac{\gamma}{g}\right) \tag{21}$$

The consistent density purpose shows using Equation (22).

$$f_{\text{Cauchy}(0,g)}(\gamma) = \frac{1}{\pi}\frac{g}{g^2 + r^2} \tag{22}$$

In Equation (22), g is a ratio limit and $\gamma = \tan(\pi(y - 1/2))$. There is a connection between the Gaussian mutation and the Cauchy function. However, the Cauchy mutation is distinct in that it is less extreme and takes a vertical form. The Cauchy mutation, on the other hand, has a larger horizontal spread than the Gaussian. The Cauchy mutation can be used to increase or decrease the number of neighbours in each generation or to alter the search capabilities of search agents.

On the other hand, this makes search agents more trustworthy in their quest to enhance the answers discovered at scale while rapidly evading the best places. The Cauchy mutation serves as a mutation operator for this purpose. The Cauchy mutation is shown by Equation (23) here.

$$x_i' = x_i \times (1 + C(\gamma)) \tag{23}$$

In Equation (23), $C(\gamma)$ is a completely arbitrary sum generated by mutation. The QLGCTSA algorithm, which incorporates the Cauchy mutation into the swarm behaviour component of the TSA algorithm, results in more misuse and more talented regions. Using the Cauchy mutation throughout the optimisation process improves solution quality overall.

4. Results and discussion

4.1. Dataset description

Four distinct public datasets are used to assess the effectiveness of the suggested approach. Having photos recorded against a variety of backdrops is why we recommend using various datasets. The backdrop in PlantVillage is predetermined and consistent. Datasets of apples, maize, and rice are collected in the wild. **Figure 2** displays several examples of photographs together with their respective classifications from each of the datasets.

(1) PlantVillage dataset—Penn State University's open dataset is frequently utilised by the scientific community for diagnosing diseases. There are 54,305 colour photos in the collection, split across 38 categories and 14 plant leaf types[37]. **Table 1** displays the total number of photos across all categories. The photographs are taken in a controlled environment with a consistent backdrop.

Table 1. PlantVillage dataset account.

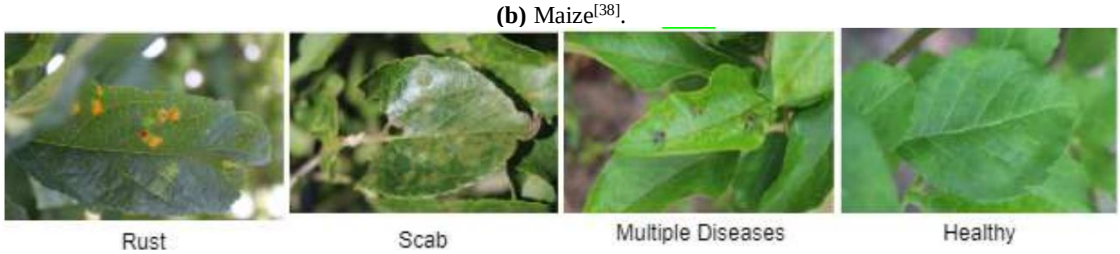
Species	Image count	Image count	Category	Species	Category
Apple	152	630	Scab	Potato	Healthy
Peach	1404	2297	Bacterial spot	Tomato	Target spot
Peach	373	360	Healthy	Tomato	Mosaic virus
Apple	1000	621	Black rot	Potato	Early scar
Apple	1000	275	Cedar apple rust	Potato	Late blight
Apple	371	1645	Healthy	Raspberry	Healthy
Cherry	5090	854	Well	Soybean	Healthy
Cherry	1835	1052	Powdery mildew	Squash	Powdery mildew
Corn	456	513	Grey leaf spot	Strawberry	Healthy
Corn	1109	1192	Mutual rust	Strawberry	Leaf scorch
Corn	2127	1162	Healthy	Tomato	Bacterial advertisement
Corn	1000	985	Northern leaf blight	Tomato	Early disfigurement
Grape	1591	1180	Black rot	Tomato	Healthy
Grape	1909	1383	Black measles	Tomato	Late disfigurement

Table 1. (*Continued*).

Species	Image count	Image count	Category	Species	Category
Grape	952	1076	Isariopsis leaf spot	Tomato	Leaf mold
Grape	1771	423	Healthy	Tomato	Septoria spot
Orange	1676	5507	Citrus greening	Tomato	Two dotted spider mite
Pepper	5357	997	Bacterial spot	Tomato	Yellow curl virus
Pepper	1502	1478	Healthy	Blueberry	Well
Total	152	-	-	-	-

(a) PlantVillage[37].

(b) Maize[38].



(c) Apple[39].

(d) Rice[38].

Figure 2. Sample images from the datasets used in the experiment.

(2) Maize dataset—In the study of Chen et al.[38], 400 photos across 4 maize disease groups make up the Maize dataset. **Table 2** provides the total number of images across all categories, including the test set. Photos are taken in their natural settings, without any manipulation of the backgrounds. There are 100 training photos in each of the categories. Some photographs from each group are pulled out and put through their paces.

Table 2. Maize dataset explanation.

Grouping	Image sum
Eyespot	121
Goss's bacterial wilting	120
Gray leaf advertisement	121
Phaeosphaeria spot	119
Total	1,821

(3) Apple dataset—Using the Kaggle platform, CVPR 2020 issued a plant pathology challenge to detect illnesses in apples[39]. There are a total of 3642 photos in the collection, split evenly across four classes; however, only 1822 of those images have been annotated. For this study, we chose labelled photos across several illness categories, including healthy, rust, scab, and others. **Table 3** lists the total number of images, broken down by kind. The photographs were taken in the field under uncontrolled lighting circumstances.

Table 3. Apple dataset report.

Image count	Type
516	Healthy
622	Rust
592	Scab
91	Multiple disease
1821	Total

(4) Rice dataset—The Fujian Institute of Subtropical Botany in Xiamen, China[38] provides this guide, which classifies rice illnesses into five broad groups. The dataset contains a total of 500 photos; 100 are assigned to each of five different categories. Different sets of test photos with varied numbers of examples are provided. **Table 4** lists the total number of images for each classification.

Table 4. Rice dataset report.

Image count	Group
107	Stackburn
115	White tip
108	Bacterial leaf line
115	Leaf injury
115	Leaf smut
560	Total

4.2. Experimental setup

The findings in this investigation are broken down into a training and testing phase. Operating workstation specs include an Intel Core i7-6800k processor, NVIDIA GTX 1080 graphics processing unit, 32 GB of RAM, and a 512 GB Samsung NVMe PCIe M2 solid state driver. Python 3.7 (Delaware, United States) and Tensorflow 2.0 (open-source artificial intelligence library) are used to build up the environment[40].

After each epoch, the model's performance is measured against the validation dataset. The model is only put into action on the test dataset when it has achieved the appropriate datasets.

All four databases were collected in various regions and provide information on several crops. Different types of datasets exist, including those that are modest and well-balanced, with 400–500 photos, and those that are vast and unbalanced, with images. The purpose of using multiple settings is to evaluate the suggested

model's efficacy in a variety of test conditions.

4.3. Visualization analysis

Figures 3–5 depict rice, apples, and maize, respectively. There are a total of 38 classifications in the PlantVillage dataset. Due to the high volume of categories, we have omitted the presentation of confusion matrices for the same. For the two classes in the Rice dataset depicted in **Figure 3**, the confusion matrix due to their similarities, a tiny percentage of samples are incorrectly labelled as either leaf scald or white tip. Similarities across illnesses may contribute to their incorrect categorization. As can be seen from the confusion matrix in **Figure 4**, several photos in the Apple dataset have numerous illness characteristics that confuse the classifier. Over 95% accuracy is reached across the board for all other categories.

The model struggled to produce satisfactory results in the tomato species on the PlantVillage dataset, particularly with regard to early spot. Likewise, the model tends to incorrectly place instances of northern leaf blight and grey spot in the same category for the corn species. Photos of eyespot and grey leaf spot illnesses in the Maize dataset are incorrectly categorised, but photos of the other two groups are correctly labelled (**Figure 5**).

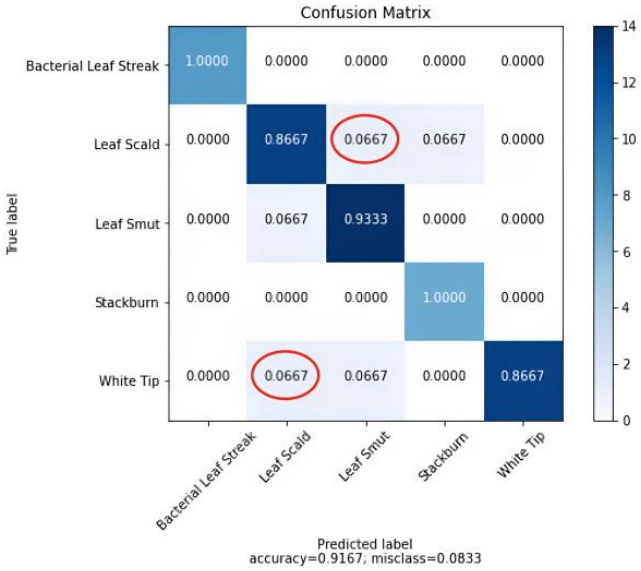


Figure 3. Confusion matrix for Rice.

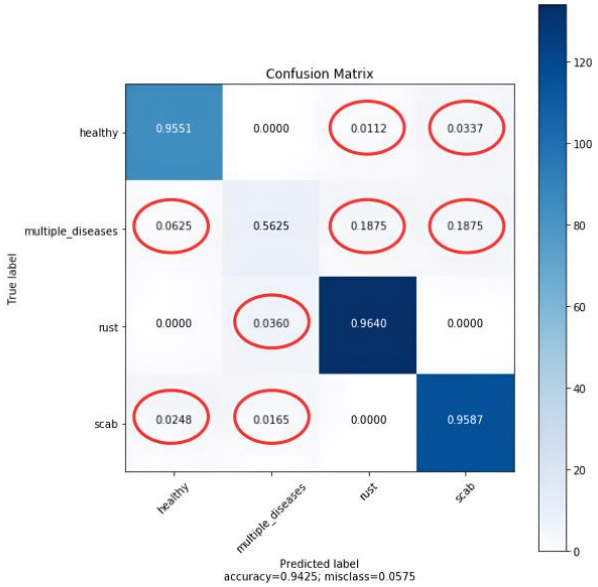


Figure 4. Confusion matrix for Apple.

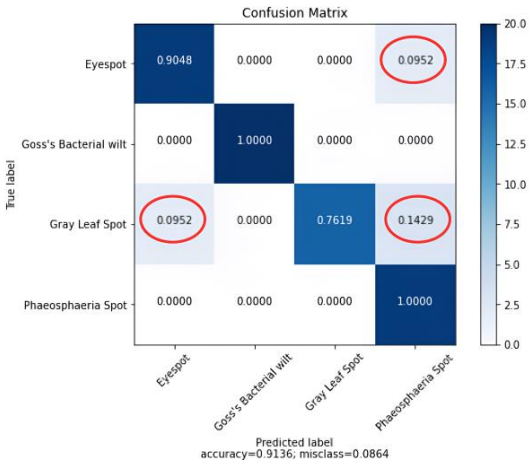


Figure 5. Confusion matrix for Maize.

4.4. Validation analysis of the proposed model

Table 5 presents the analysis of a proposed model for all datasets in terms of various metrics.

Table 5. Performance analysis of proposed model.

Measures	Precision	Sensitivity	Specificity	Accuracy	F-score	Kappa
Rice dataset	0.9984	0.9981	0.9984	0.9983	0.9983	0.9966
Apple dataset	0.9972	0.9969	0.9972	0.9971	0.9971	0.9941
Maize dataset	0.9963	0.9953	0.9963	0.9958	0.9958	0.9916
PlantVillage	0.9973	0.9968	0.9973	0.9971	0.9971	0.9941

The proposed model's performance analysis is shown in **Table 5** above. In this analysis, we used different datasets to evaluate the results. The Rice dataset had the precision of 0.9984, sensitivity rate of 0.9981, specificity of 0.9984, accuracy of 0.9983, and kappa range of 0.9966, all of which were achieved respectively. Then, another dataset known as the Apple dataset achieved precision of 0.9972, sensitivity rate of 0.9969, specificity of 0.9972, accuracy of 0.9971, F-score of 0.9971, and finally kappa range of 0.9941, all in accordance with each other. Another dataset, the Maize dataset, then achieved precision of 0.9963, sensitivity rate of 0.9953, specificity of 0.9963, and finally kappa range of 0.9916, all in accordance with each other. Another dataset, called PlantVillage, then achieved precision of 0.9973, sensitivity rate of 0.9968, specificity of 0.9973, accuracy of 0.9971, and finally kappa range of 0.9941, all in accordance with each other.

4.5. Comparative analysis of various training and testing data

Tables 6 and **7** provide the comparative analysis of the proposed model with existing techniques for 60%–40% and 80%–20%. The existing models, such as VGG-ICNN[21], U-Net[22], EfficientNet[23], GoogleNet[28] and DenseNet[32], are considered for validation, and results are averaged in **Tables 6** and **7**.

Table 6. Comparative analysis of the proposed model on 60%–40%.

Models	Sensitivity	Specificity	F-measure	Accuracy
U-Net	75.00	80.00	76.92	77.50
EfficientNet	85.00	75.00	80.95	80.00
GoogleNet	80.00	70.00	76.19	75.00
DenseNet	90.00	80.00	85.71	85.00
VGG-ICNN	90.00	85.00	87.80	87.50
Proposed	95.36	90.00	95.24	96.80

Table 6 above shows the comparative analysis of the suggested model on a 60%–40% basis. In the analysis of the U-Net model, the sensitivity, specificity, F-measure range, and accuracy rate were all reached at 75.00, 80.00, and 76.92, respectively. After that, the EfficientNet model achieved the following results: sensitivity of 85.00, specificity of 75.00, F-measure range of 80.95, and accuracy rate of 80.00, respectively. Then, the GoogleNet model achieved the following values: sensitivity: 80.00; specificity: 70.00; F-measure range: 76.19; and accuracy rate: 75.00, respectively. Then, the DenseNet model achieved the following values: sensitivity = 90.00; specificity = 80.00; f-measure range = 85.71; and accuracy rate = 85.00. Then, the sensitivity, specificity, F-measure range, and accuracy rate of the VGG-ICNN model were all 90.00, 85.00, and 87.80, respectively. The proposed model then achieved the following results: sensitivity of 95.36, specificity of 90.00, F-measure range of 95.24, and accuracy rate of 96.80, respectively.

The comparative analysis of the proposed model on 80%–20% is shown in **Table 7**. In the analysis, we used various models, including the U-Net 93.61, the 94.56 specificity, the 93.20 f-measure range, and the 94.08 accuracy rate. Then, the EfficientNet model achieved the following results: sensitivity of 91.00; specificity of 93.00; F-measure range of 95.00; and accuracy rate of 92.29, respectively. Then, the GoogleNet model achieved sensitivity, specificity, and accuracy rates of 94.00, 96.00, and 96.14, respectively. Then the sensitivity, specificity, f-measure range, and accuracy rate of the DenseNet model were each 95.00, 96.00, and 97.00, respectively. The VGG-ICNN model then achieved the sensitivity, specificity, f-measure range, and accuracy rate of 98.00, 98.00, and 98.14, respectively. The proposed model then achieved the following values for sensitivity, specificity, F-measure range, and accuracy rate: 99.73, 99.68, 99.71, and 99.71 respectively. **Figures 6–9** present the visual representation of proposed model with existing techniques for different ratio of training and testing data.

Table 7. Comparative analysis of the proposed model on 80%–20%.

Models	Sensitivity	Specificity	F-Score	Accuracy
U-Net	93.61	94.56	93.20	94.08
EfficientNet	91.00	93.00	95.00	92.29
GoogleNet	94.00	94.00	96.00	96.14
DenseNet	95.00	96.00	97.00	97.21
VGG-ICNN	98.00	98.00	98.00	98.14
Proposed	**99.73**	**99.68**	**99.71**	**99.71**

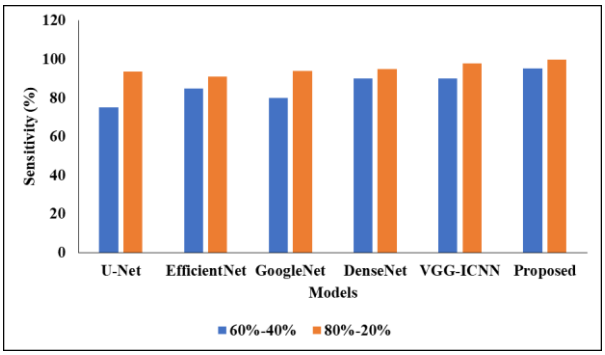


Figure 6. Comparative analysis of the proposed model.

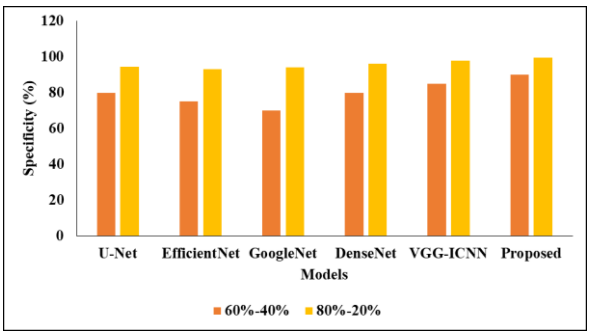


Figure 7. Graphical representation of various DL models.

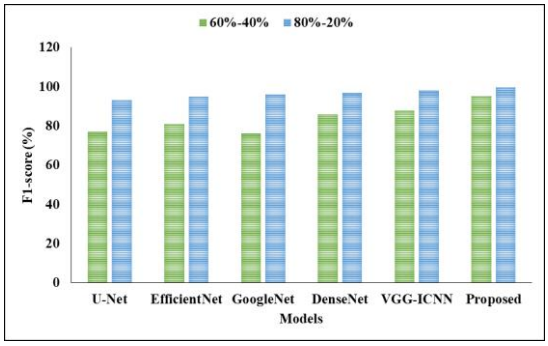


Figure 8. F-score analysis.

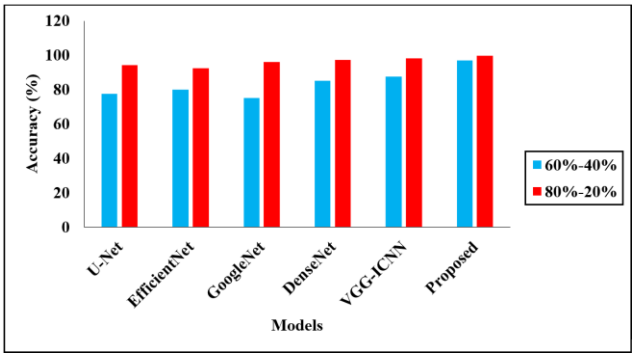


Figure 9. Accuracy analysis of various training and testing data.

5. Conclusion and future work

Diseases that are detected and treated early on have the greatest impact on crop yields and quality. Improvements in performance and applicability to the creation of portable IoT devices for smart agriculture are possible thanks to the mechanisation of the disease detection work utilising image processing, learning. Preliminary results from using deep learning models for agricultural disease diagnosis have been encouraging, especially from using CNNs. However, the usefulness is constrained by the high memory and processing needs of deep CNN models. Lightweight CNN representations have inconsistent performance across plant disease kinds since they are only equipped to deal with a limited number of disease categories. The research presented here suggests using a GRMA-Net to categorise diseases. Our GRMA-Net is able to extract discriminative features by focusing on revealing regions at many scales thanks to its multiscale attention module. In addition, our GRMA-Net employs GRUs to more effectively leverage the interdependence and contextual interaction between variables in different geographic areas. The suggested model is evaluated on four datasets, where it outperforms the state-of-the-art models by a wide margin (between 95% and 99%). The suggested model consistently beats state-of-the-art lightweight approaches and most recently announced deep CNN models across all four available datasets.

Future works

Multi-sensor data fusion from UAVs, satellite and UAV image fusion, and multi-resolution data fusion are the most common types of fusion utilised in agriculture. In applications like crop monitoring and plant categorization, this fusion is used to enhance the detection procedure. As a result, including more data sources can improve the efficiency with which we diagnose diseases in their earliest stages. However, there has been a lack of research on multimodal fusion, which is especially lacking in the field of illness diagnosis. In this publication, we showed encouraging multimodal fusion findings, showing the great promise of data. This opens the door to other studies.

Author contributions

Conceptualization, BR and SKB; methodology, BSKD; software, BSKD; validation, BPK and GHS; formal analysis, BR; investigation, SKB; resources, BSKD; data curation, BPK; writing—original draft preparation, BR and SKB; writing—review and editing, BSKD; visualization, BPK; supervision, GHS; project administration, BPK and GHS; funding acquisition, GHS. All authors have read and agreed to the published version of the manuscript.

Conflict of interest

The authors declare no conflict of interest.

References

1. Kolhe P, Kalbande K, Deshmukh A. Internet of Thing and Machine Learning Approach for Agricultural Application: A Review. 2022 10th International Conference on Emerging Trends in Engineering and Technology - Signal and Information Processing (ICETET-SIP-22). Published online April 29, 2022. doi: 10.1109/icetet-sip-2254415.2022.9791751
2. Thakur PS, Khanna P, Sheorey T, et al. Trends in vision-based machine learning techniques for plant disease identification: A systematic review. Expert Systems with Applications. 2022, 208: 118117. doi: 10.1016/j.eswa.2022.118117
3. Jackulin C, Murugavalli S. A comprehensive review on detection of plant disease using machine learning and deep learning approaches. Measurement: Sensors. 2022, 24: 100441. doi: 10.1016/j.measen.2022.100441
4. Liu Z, Bashir RN, Iqbal S, et al. Internet of Things (IoT) and Machine Learning Model of Plant Disease Prediction–Blister Blight for Tea Plant. IEEE Access. 2022, 10: 44934-44944. doi: 10.1109/access.2022.3169147
5. Li L, Zhang S, Wang B. Plant Disease Detection and Classification by Deep Learning—A Review. IEEE Access. 2021, 9: 56683-56698. doi: 10.1109/access.2021.3069646
6. Sujatha R, Chatterjee JM, Jhanjhi N, et al. Performance of deep learning vs machine learning in plant leaf disease detection. Microprocessors and Microsystems. 2021, 80: 103615. doi: 10.1016/j.micpro.2020.103615
7. Jogekar RN, Tiwari N. A Review of Deep Learning Techniques for Identification and Diagnosis of Plant Leaf Disease. Smart Innovation, Systems and Technologies. Published online July 18, 2020: 435-441. doi: 10.1007/978-981-15-5224-3_43
8. Delnevo G, Girau R, Ceccarini C, et al. A Deep Learning and Social IoT Approach for Plants Disease Prediction Toward a Sustainable Agriculture. IEEE Internet of Things Journal. 2022, 9(10): 7243-7250. doi: 10.1109/jiot.2021.3097379
9. Bi C, Wang J, Duan Y, et al. MobileNet based apple leaf diseases identification. Mobile Networks and Applications. 2022, pp. 1-9.
10. Lakshmanarao A, Babu MR, Kiran TSR. Plant Disease Prediction and classification using Deep Learning ConvNets. 2021 International Conference on Artificial Intelligence and Machine Vision (AIMV). Published online September 24, 2021. doi: 10.1109/aimv53313.2021.9670918
11. Ahmed AA, Reddy GH. A Mobile-Based System for Detecting Plant Leaf Diseases Using Deep Learning. AgriEngineering. 2021, 3(3): 478-493. doi: 10.3390/agriengineering3030032
12. Kundu N, Rani G, Dhaka VS, et al. IoT and Interpretable Machine Learning Based Framework for Disease Prediction in Pearl Millet. Sensors. 2021, 21(16): 5386. doi: 10.3390/s21165386
13. Islam MdA, Nymur Md, Shamsojjaman M, et al. An Automated Convolutional Neural Network Based Approach for Paddy Leaf Disease Detection. International Journal of Advanced Computer Science and Applications. 2021, 12(1). doi: 10.14569/ijacsa.2021.0120134
14. Kabir MM, Ohi AQ, Mridha MF. A Multi-Plant Disease Diagnosis Method Using Convolutional Neural Network. Computer Vision and Machine Learning in Agriculture. Published online 2021: 99-111. doi: 10.1007/978-981-33-

6424-0_7
15. Gonzalez-Huitron V, León-Borges JA, Rodriguez-Mata AE, et al. Disease detection in tomato leaves via CNN with lightweight architectures implemented in Raspberry Pi 4. Computers and Electronics in Agriculture. 2021, 181: 105951. doi: 10.1016/j.compag.2020.105951
16. Latha RS, Sreekanth GR, Suganthe RC, et al. Automatic Detection of Tea Leaf Diseases using Deep Convolution Neural Network. 2021 International Conference on Computer Communication and Informatics (ICCCI). Published online January 27, 2021. doi: 10.1109/iccci50826.2021.9402225
17. Applalanaidu MV, Kumaravelan G. A Review of Machine Learning Approaches in Plant Leaf Disease Detection and Classification. 2021 Third International Conference on Intelligent Communication Technologies and Virtual Mobile Networks (ICICV). Published online February 4, 2021. doi: 10.1109/icicv50876.2021.9388488
18. Zhou C, Zhou S, Xing J, et al. Tomato Leaf Disease Identification by Restructured Deep Residual Dense Network. IEEE Access. 2021, 9: 28822-28831. doi: 10.1109/access.2021.3058947
19. Bedi P, Gole P. Plant disease detection using hybrid model based on convolutional autoencoder and convolutional neural network. Artificial Intelligence in Agriculture. 2021, 5: 90-101. doi: 10.1016/j.aiia.2021.05.002
20. Roy AM, Bhaduri J. A Deep Learning Enabled Multi-Class Plant Disease Detection Model Based on Computer Vision. AI. 2021, 2(3): 413-428. doi: 10.3390/ai2030026
21. Thakur PS, Sheorey T, Ojha A. VGG-ICNN: A Lightweight CNN model for crop disease identification. Multimedia Tools and Applications. 2022, 82(1): 497-520. doi: 10.1007/s11042-022-13144-z
22. Divyanth LG, Ahmad A, Saraswat D. A two-stage deep-learning based segmentation model for crop disease quantification based on corn field imagery. Smart Agricultural Technology. 2023, 3: 100108. doi: 10.1016/j.atech.2022.100108
23. Nigam S, Jain R, Marwaha S, et al. Deep transfer learning model for disease identification in wheat crop. Ecological Informatics. 2023, 75: 102068. doi: 10.1016/j.ecoinf.2023.102068
24. Zhao Y, Yang Y, Xu X, et al. Precision detection of crop diseases based on improved YOLOv5 model. Frontiers in Plant Science. 2023, 13. doi: 10.3389/fpls.2022.1066835
25. Sanida MV, Sanida T, Sideris A, et al. An Efficient Hybrid CNN Classification Model for Tomato Crop Disease. Technologies. 2023, 11(1): 10. doi: 10.3390/technologies11010010
26. Xu L, Cao B, Zhao F, et al. Wheat leaf disease identification based on deep learning algorithms. Physiological and Molecular Plant Pathology. 2023, 123: 101940. doi: 10.1016/j.pmpp.2022.101940
27. Zhang R, Wang Y, Jiang P, et al. IBSA_Net: A Network for Tomato Leaf Disease Identification Based on Transfer Learning with Small Samples. Applied Sciences. 2023, 13(7): 4348. doi: 10.3390/app13074348
28. Joshi BM, Bhavsar H. Deep Learning Technology based Night-CNN for Nightshade Crop Leaf Disease Detection. International Journal of Intelligent Systems and Applications in Engineering. 2023, 11(1), 215-227.
29. Ma W, Yu H, Fang W, et al. Crop Disease Detection against Complex Background Based on Improved Atrous Spatial Pyramid Pooling. Electronics. 2023, 12(1): 216. doi: 10.3390/electronics12010216
30. Sanida T, Sideris A, Sanida MV, et al. Tomato leaf disease identification via two–stage transfer learning approach. Smart Agricultural Technology. 2023, 5: 100275. doi: 10.1016/j.atech.2023.100275
31. Si X, Hong B, Hu Y, et al. Crop Disease Recognition Based on Improved Model-Agnostic Meta-Learning. Computers, Materials & Continua. 2023, 75(3): 6101-6118. doi: 10.32604/cmc.2023.036829
32. Jiang M, Feng C, Fang X, et al. Rice Disease Identification Method Based on Attention Mechanism and Deep Dense Network. Electronics. 2023, 12(3): 508. doi: 10.3390/electronics12030508
33. Shah A, Bangash JI, Khan AW, et al. Comparative analysis of median filter and its variants for removal of impulse noise from gray scale images. Journal of King Saud University - Computer and Information Sciences. 2022, 34(3): 505-519. doi: 10.1016/j.jksuci.2020.03.007
34. Ramamoorthy M, Qamar S, Manikandan R, et al. Earlier Detection of Brain Tumor by Pre-Processing Based on Histogram Equalization with Neural Network. Healthcare. 2022, 10(7): 1218. doi: 10.3390/healthcare10071218
35. Abernathy A, Celebi ME. The incremental online k-means clustering algorithm and its application to color quantization. Expert Systems with Applications. 2022, 207: 117927. doi: 10.1016/j.eswa.2022.117927
36. Kaur S, Awasthi LK, Sangal AL, et al. Tunicate Swarm Algorithm: A new bio-inspired based metaheuristic paradigm for global optimization. Engineering Applications of Artificial Intelligence. 2020, 90: 103541. doi: 10.1016/j.engappai.2020.103541
37. Hughes D, Salath´e M. An open access repository of images on plant health to enable the development of mobile disease diagnostics, arXiv preprint arXiv:1511.08060 (2015).
38. Chen J, Chen J, Zhang D, et al. Using deep transfer learning for image-based plant disease identification. Computers and Electronics in Agriculture. 2020, 173: 105393. doi: 10.1016/j.compag.2020.105393
39. Thapa R, Snavely N, Belongie S, Khan A. The plant pathology 2020 challenge dataset to classify foliar disease of apples, arXiv preprint arXiv:2004.11958 (2020).
40. Thirumalraj, A., Anusuya, V. S., & Manjunatha, B. Detection of Ephemeral Sand River Flow Using Hybrid Sandpiper Optimization-Based CNN Model. In Innovations in Machine Learning and IoT for Water Management IGI Global. pp. 195-214, 2024. doi: 10.4018/979-8-3693-1194-3.ch010