EASR

Engineering and Applied Science Research

https://www.tci-thaijo.org/index.php/easr/index

Published by the Faculty of Engineering, Khon Kaen University, Thailand

IDRCNN and BDC-LSTM: An efficient novel ensemble deep learning-based approach for accurate plant disease categorization

Shaik Salma Asiya Begum* and Hussain Syed

School of Computer Science and Engineering, VIT-AP University, Amaravati, Andhra Pradesh, 522237, India

Received 1 May 2024
Revised 12 December 2024
Accepted 25 December 2024

Abstract

Agriculture provides food for everyone, even as the population grows. Early detection of plant illnesses in the agricultural industry is recommended to guarantee enough food for everybody. However, early crop disease detection is impossible. Nevertheless, overfitting is a common problem with these studies, and using test datasets from unfamiliar regions substantially decreases diagnostic performance. This study proposes a new Deep Learning (DL) approach for categorising illnesses affecting plant leaves. The limited number of plant leaf illness datasets can result in inadequate and overfitting generalisation of ML approaches. The use of augmentation contributes to the artificial increase in dataset variation. Remove unnecessary noise, improve the image's contrast, and eliminate the background during pre-processing. Next, the Res2Net technique extracts the appropriate attributes from the picture, including structure, colour, and texture elements. Next, hybrid deep learning (DL) approaches of the Bi-Directional Convolutional Long Short-Term Memory (BDC-LSTM) and Improved Deep Residual Convolutional Neural Network (IDRCNN) categorise plant illness into their categories. Finally, the Enhanced Watershed Segmentation Algorithm (EWSA) will extract the disease portion from the input picture. We utilised a dataset on leaf diseases for model simulation and assessment. Experimental results show that the proposed method is more accurate in classifying and identifying plant leaf disease than other approaches. The uses of the proposed DL method show their efficiency and excellent accuracy.

Keywords: Classification, Enhanced watershed segmentation algorithm, Feature extraction, Improved deep residual convolutional neural network, Plant leaf disease, Pre-processing

1. Introduction

Plant diseases affect the roots, leaves, fruit, and stems. It also affects the number and quality of crops, exacerbating the world's food insecurity and poverty. Agricultural disease is the primary source of food shortages and increased food production costs; it is estimated that global agrarian output loss will be 16% annually [1-3]. Biologic factors include bacteria, fungi, and algae-caused diseases; biotic factors that cause diseases include rainfall, humidity, temperature, and nutrient deficiencies.

One significant contributing cause of crop loss is plant diseases. They could degrade crop quality and lower agricultural productivity. India is one of the primary occupations in developing countries in agriculture. The country's economy may suffer due to crop losses [4, 5]. Crops are affected by many diseases, particularly in temperate, tropical, and subtropical areas. Next, plant infections can be caused by specific moulds, viruses, bacteria, fungi, and occasionally environmental factors, including humidity, temperature, and precipitation. After that, these substances may serve as a conduit for the spread of viruses, epidemics and other illnesses. These diseases could severely affect farmers' livelihoods and cause significant financial losses [6-8]. Early detection of plant diseases is essential for crop control and management. Moreover, farmers and field experts use traditional methods to detect plant illnesses, which are expensive, time-consuming, and error-prone.

Ecological, economic, and social elements of agriculture are all impacted by plant illnesses, which also affect plant growth and agricultural yield [9, 10]. New studies on leaf illnesses elucidate plant damage. Plant leaf illnesses are also quite costly for farmers. Early illness detection needs to be given special attention. The biological causes of plant illnesses are the main focus of the literature. Their conclusions are based on what is visible on the surfaces of the plants and leaves. Early problem discovery is a crucial first step towards optimal disease management [11, 12]. In the past, detection was performed by specialists with human experience. Although human specialists possess visual illness identification skills, certain obstacles may compromise effectiveness. In this case, it is critical to quickly and precisely diagnose and classify illnesses.

AI-based research advances have automatically made it possible to identify plant illnesses from simple pictures. A benefit of DL is its ability to extract features from pictures [13] automatically. CNN is a multi-layer feed-forward neural network and a popular DL model. Researchers and scientists recognise the crucial issues and constraints while examining plant leaf diseases [14, 15]. These include the need for high-quality leaf images, the possibility of noisy data affecting the leaf samples, the potential of environmental influences causing changes in leaf colour, and the possibility of different illnesses affecting different plants, making disease identification difficult. Most of the methods that were made use of controlled datasets, which are collections of pictures taken under

*Corresponding author.
Email address: shaiksalmaasiyabegum@gmail.com
doi: 10.14456/easr.2025.3

ideal circumstances in a regulated setting. For potential plant leaf disease identification and categorisation, it is not possible to get high-resolution, high-quality images in the actual world. To address these problems, we propose a novel DL-based method for classifying plant leaf diseases. In our manuscript, we focus on addressing several key challenges observed in previous works on plant leaf disease detection. Many earlier studies struggled with overfitting due to limited or imbalanced datasets, which we mitigate by leveraging data augmentation through MCGAN, enhancing the model's generalisation. Additionally, we addressed the common issue of poor input image quality by implementing advanced preprocessing techniques like noise reduction and contrast enhancement, leading to more accurate disease classification. To improve feature extraction, which is crucial for identifying disease-specific characteristics such as colour, shape, and texture, we employed the Res2Net method. Furthermore, to overcome the limitations of single-model approaches, we integrated BDC-LSTM and IDRCNN models, creating a hybrid deep-learning framework that enhances classification accuracy. Finally, to ensure precise segmentation of diseased regions, we used EWSA, enabling more precise visualisation and more targeted analysis. These innovations collectively aim to advance the field by overcoming the limitations of prior research, resulting in a more robust and accurate plant leaf disease detection system.

1.1 The novelty of this research

- By leveraging data augmentation techniques, the artificial diversity of the training dataset is increased, addressing the issue of overfitting and enhancing the generalisation capability of machine learning algorithms for plant leaf disease detection.
- Implementing advanced preprocessing methods such as noise reduction, contrast enhancement, and background elimination ensures higher-quality input images, which are crucial for accurate disease classification.
- The utilisation of the Res2Net method enables the effective extraction of critical image features, including colour, shape, and texture, which are essential for precise disease identification.
- The integration of BDC-LSTM and IDRCNN models offers a robust hybrid deep-learning approach for categorising plant diseases into specific categories, improving classification accuracy.
- The use of EWSA for segmenting diseased regions within the input image allows for clear visualisation of the affected areas, facilitating more targeted analysis and intervention.

Section 2 describes the details of the latest studies for leaf diseases. The proposed model for detecting leaf disease is covered in Section 3. In Section 4, tomato leaf samples with six different illnesses are used to examine the results of the proposed structure. The model for leaf recognition is concluded in Section 5.

1.2 Literature survey

Over the past few decades, many of the researchers have worked on crops other than plant leaves; they were not limited to studying crop diseases that impact plant leaves. An automated model that recognises and classifies plant leaf diseases using an OMNCNN was presented by Abbas et al. [16]. The affected regions of the leaf picture are identified using bilateral filtering-based (BF) pre-processing and Kapur's thresholding-based segmentation technique. Additionally, the MobileNet model's hyper-parameters are adjusted using the Emperor Penguin Optimizer (EPO) approach as a feature extraction technique to raise the detection rate of plant illnesses. An ELM-based classifier were finally used to assign the precise class labels to the applied plant leaf images. For training purposes, they generated artificial photos using the C-GAN and added them to the dataset in Ashwinkumar et al. [17]. After that, a DenseNet121 approach trained via transfer learning on synthetic and real-world images was used to categorise the tomato leaf images into ten different disease kinds.

An improved recognition and categorisation method based on DL models was suggested by Elaraby et al. [18] to address the issue of disease recognition on the three primary citrus diseases: canker, Huanglongbing, and black spot. Sharing attributes reduces training overhead between the classifier and RPN. Without encountering gradient vanishing problems, the remaining connections enable the model to grow deeper and improve accuracy. The recommended model uses ResNet101 for collecting attributes.

A DL model for categorising and diagnosing plant leaf diseases was suggested by Umamageswari et al. [19]. Data augmentation was believed to help with training on a plant illness classification challenge. This method artificially boosts the quantity of training pictures by using label-preserving alterations. A deep neural network can extract features from coloured, raw pictures. AlexNet is a framework that uses transfer learning to be trained to obtain characteristics from inputs. PSO is used with the 100 characteristics that AlexNet has collected to choose characteristics. They recommended that each feature be represented in PSO, utilising a binary selection for feature selection. The technique artificially increases the number of training images via label-preserving modifications. When attempting to obtain characteristics from coloured, plain images, a deep neural network comes in helpful.

Syed-Ab-Rahman et al. [20] created a DNN to classify rice leaf diseases. Optimising weights and biases to minimise errors is the goal of CSA, a metaheuristic search technique based on crow behaviour. The standard pre-tuning and fine-training procedures were optimised to create a DNN-CSA structure. After pre-processing the pictures of paddy leaves, regions suggestive of illness were extracted using the k-means clustering method. After that, healthy areas were eliminated using thresholding. Then, properties, including colour, texture, and shape, were extracted from the previously divided impacted regions. Eventually, the proposed method was used to classify paddy leaf illnesses.

A technique for diagnosing plant leaf illnesses has been suggested by Nalini et al. [21]. First, unwanted noise and overfitting are removed, increasing the picture's contrast. Second, the Chameleon Swarm Algorithm (CSA), or FCM-CSA, is based on fuzzy C-Means (FCM) and used in the plant leaf sick section segmentation process. In the third phase, features are extracted using a fast GLCM retrieval of characteristics model. The Progressive Neural Architecture Search (PNAS) is utilised to identify plant leaf illnesses.

An enhanced CLGAN (Crop Leaf GAN) model with an improved loss function was developed by Sharma et al. [22], significantly reducing the number of parameters compared to previous state-of-the-art approaches. To mitigate issues such as vanishing gradients, training instability, and non-convergence, the generator and discriminator in the CLGAN model were incorporated into an encoder-decoder network, enabling the preservation of complex details during synthetic image generation with clear lesion differentiation. The proposed loss function introduces a dynamic correction factor, promoting learning stability while ensuring effective weight optimisation.

The SoyaTrans network, a robust model introduced by Sharma et al. [23], integrates CNN architecture with Swin Transformers to process real-world field images effectively. Unlike the conventional cyclic shift, this model employs a random shifting technique,

which improves classification performance while reducing computational complexity. Furthermore, SoyaTrans combines the strengths of a traditional CNN with a Swin Transformer, enabling efficient disease detection across various crop types.

Different crops can exhibit diverse variations and are susceptible to various diseases, which has been the issue statement of prior articles. Creating a comprehensive framework for identifying different crop species' illnesses is quite time-consuming. Large agricultural fields need time to inspect for illness manually. Early identification is crucial to lessen the consequences of diseases and prevent them from spreading to other plants or crops. However, mild symptoms may go unnoticed in the initial phases of a disease, making it challenging to respond when necessary.

2. Materials and methods

Illnesses of the leaves harm the health and activity of the plant and often appear on the leaves. Environmental stresses like nutrient shortages, pollution, drought, and infections brought on by bacteria, fungi, nematodes, and viruses can all contribute to these illnesses. This work offers a new DL-based plant leaf illness diagnosis and classification method. Plant leaf illnesses often have few large datasets available, which can lead to overfitting and insufficient generalisation of ML algorithms. Data augmentation can intentionally be used to improve the training dataset's variability. During the pre-processing phase, take the following actions: remove any unwanted noise, increase the contrast of the picture, and eliminate the background. Next, the essential attributes from the image are retrieved using the Res2Net method, including the colour, texture, and form elements. Next, plant illnesses are categorised using hybrid DL techniques that use the IDRCNN and BDC-LSTM. Finally, the afflicted region is separated from the input picture using the EWS method. It used a leaf disease dataset to simulate and evaluate the model. Figure 1 shows the architecture of the proposed system.

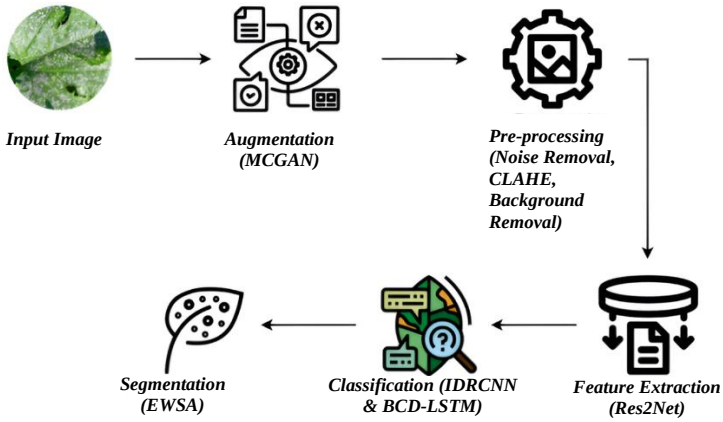


Figure 1 The structure of the proposed model.

2.1 Data augmentation

The colour images from the Plant Village dataset were used during this study. The dataset included 14 crop species and 26 illnesses distributed among 38 groups. Unlike other types of image data, plant leaf data, especially from datasets such as PlantVillage, exhibits various distinct features. The wide variations in leaf colour, texture, and shape that result from different plant species, growth phases, and environmental circumstances are among them. Furthermore, it might be difficult to accurately extract and classify features from plant leaves due to their ability to display minute changes between unhealthy and healthy states. One of the main issues with the dataset under study is the coexistence of similar visual symptoms across several diseases, which could result in incorrect classification. Additional variables in the dataset that may impact the model's generalisation capacity include changes in background noise, leaf orientations, and lighting. Furthermore, training challenges may arise from the uneven distribution of samples among various illness classes, which could lead to the model being biased towards more common courses. To address this, a set of representative images generated by the Modified Conditional Generative Adversarial Network (MCGAN) was used in this study. This will provide visual insights into the variations and improvements introduced by the GAN, especially in enhancing data diversity for improved model training and robustness in disease classification. Including these images will also help readers understand how GAN-generated samples contribute to the overall effectiveness of the proposed model. To overcome these obstacles, we used sophisticated feature extraction techniques like Res2Net and data augmentation approaches to increase the precision and resilience of the illness prediction model. Furthermore, an issue which frequently comes with a large dataset needs to be more balanced. Overfitting regularisation techniques were developed to counter this, such as data augmentation after pretreatment. The pictures were rotated clockwise and counterclockwise, resized, flipped vertically and horizontally, and had their zoom intensified, among other augmentation procedures. As a result of the pictures being somewhat altered rather than replicated through the training phase, the physical files containing the augmented images were only used temporarily. This augmentation method keeps the model from overfitting and changing its shape. It strengthens its resilience, increasing the model's capacity to classify images of plant diseases in the real world reliably. During the model's training stage, data augmentation techniques can reduce overfitting by increasing the size of the dataset and adding a restricted quantity of distorted images to the training set. Because of the augmentation techniques, the dataset's size increased from 55,448 to 234,008 images. To create a dataset that is specially modified to the intricacies and differences in our study, we decided to employ a Modified Conditional Generative Adversarial Network (MCGAN) for data augmentation. PlantVillage's enhanced version is a helpful tool. Still, MCGAN's augmentation technique is more structured and customisable, producing artificial images that closely resemble differences in plant leaf conditions found in the actual world. With this method, we can add particular challenges that might be partially reflected in the present augmented dataset, like different lighting situations, angles, and leaf abnormalities. Moreover, the flexibility to increase the diversity and richness of the training data is provided by employing MCGAN for augmentation, which may improve the generalisation and resilience of our predictive model. The objective is to advance illness detection capabilities and make a significant contribution to the field by developing a custom-augmented dataset.

2.1.1 Modified conditional GAN

The GAN is one DL approach for managing complex, real-time data on a large. The GAN was combined with the Discriminator ($\mathfrak{D}$) and Generator ($\mathcal{g}$) neural networks [24]. The weight reduction ends when the categorized outcome goes back to G and $\mathfrak{D}$. The procedure is repeated until $\mathfrak{D}$ can correctly classify both produced and original image samples.

$$\min_{\mathcal{g}} \max_{\mathfrak{D}} V(\mathcal{g}, \mathfrak{D}) = \mathbb{E}_{s\sim p(s)}[\log \mathfrak{D}(s)] + \mathbb{E}_{z\sim p(z)}[\log(1 - \mathfrak{D}(\mathcal{g}(z))] \quad (1)$$

This distributes actual data occurrences in $p(s)$. More information is required for the $\mathfrak{D}$s samples than the $\mathfrak{D}(\mathcal{g}(z))$ samples to differentiate between produced and accurate information. A drawback of the conventional GAN approach is the occurrence of mode-collapse. This issue may arise in practice from a multi-modal sample distribution.

A modified form of the GAN technique called the conditional Generative Adversarial Network (CGAN) was created to solve the issue above. When used in tandem with the current distribution samples, CGAN facilitates learning effectively.

$$\min_{\mathcal{g}} \max_{\mathfrak{D}} V(\mathcal{g}, \mathfrak{D}) = \mathbb{E}_{s\sim p(s)}[\log \mathfrak{D}(s|x)] + \mathbb{E}_{z\sim p(z)}[\log(1 - \mathfrak{D}(\mathcal{g}(z|x))] \quad (2)$$

Class disparity issues are reduced, and GAN and CGAN provide a set of instances where x represents the variables specified in Eq. (1) and the class data. However, their Jensen-Shannon distributer method involves an unsuitable or irrational overlay between the distribution of actual and fictitious examples.

This study improves CGAN by integrating the Lipschitz border and the Wasserstein distance to address the abovementioned problems.

$$V(\mathcal{g}, \mathfrak{D}) = \max_{\mathfrak{D}}\{\mathbb{E}_{s\sim p(s)}[\mathfrak{D}(s|x)] - \mathbb{E}_{s\sim p(g)}[\mathfrak{D}(s|x)] - \varphi\mathbb{E}_{s\sim p(\omega)}[\|\nabla_s \mathfrak{D}(s|x)\| - 1]^2\} \quad (3)$$

2.2 Pre-Processing

Pre-processing involves removing the background, enhancing contrast, and reducing noise from the input picture after augmentation. Pre-processing consists of eliminating noise and variations from the photos to improve image quality. Mean filtering was used as the method. The filters created masks covering every stream pixel; the elements were then averaged to make an exclusive pixel.

To improve contrast, apply the Contrast-Limited Adaptive Histogram Equalization (CLAHE) technique. Two noteworthy benefits of CLAHE over conventional histogram equalisation techniques. First, all pixels should be brighter and more uniform due to CLAHE. CLAHE enhances contrast throughout the image by improving it in each zone.

$$D_B = \frac{255}{8\times 8}\sum_{i=0}^{D_A} H(i) \quad (4)$$

Secondly, CLAHE could reduce contrast enhancement, therefore reducing the issue of noise amplification. To retain the similarity and improve the entire histogram region, the histogram grows to an L height. The modified, final histogram looks like this:

$$H(i) == \begin{cases} H(i) + L, \ \mathrm{H}(i) < H_{max} \\ H \ \mathrm{H}(i)_{max_{max}}\{ \end{cases} \quad (5)$$

Pictures are first converted from RGB to HSV colour space to eliminate backgrounds. The saturation of a leaf is not the same as that of a background. Binary thresholding is applied to images on their saturation level. The number 71 has been chosen as the background separation threshold. The background-free image is then obtained by masking the original photo with the thresholded image. The pre-processing stage performance of the approaches is displayed in Figure 2.

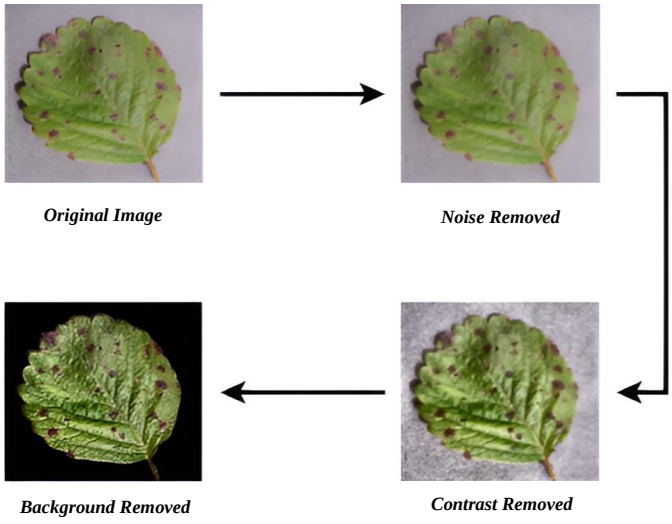


Figure 2 Pre-processing performance on pictures.

2.3 Feature extraction

To increase classification accuracy and lower model complexity, the researchers worked to reduce the amount of attributes utilising the same dataset. It has conducted a thorough analysis of the features of diseases of leaves. Finally, it selected a group of 26 characteristics from the texture, colour, and shape domain that can accurately describe disease-damaged leaf patches. These are covered in the following section. In the proposed methodology, Res2Net is employed as an additional feature extraction step to capture intricate image details, such as colour, shape, and texture, which are crucial for accurate plant leaf disease identification. To assess the necessity of this step, we could measure the model's performance by evaluating its accuracy when relying solely on the deep learning model without explicit feature extraction. This comparison would highlight the value of incorporating Res2Net in enhancing the model's predictive capabilities.

This method increases receptive fields and extracts feature power without sacrificing information. First, p subsets of the 3×3 convolution layer are equally divided, utilising the formula. $x = \{x_1, x_2, x_3, \dots, x_p\}$. Next, $Conv_q$, a 3×3 convolution receives input from all subgroups (except from x_1). The processing formula for the Res2Net [25] module can be expressed as follows:

$$g_q = \begin{cases} x_q & q = 1; \\ Conv_q(x_q) & q = 2; \\ Conv_q(x_q + g_{q-1}), & q \geq 3, q \text{ is an integer} \end{cases} \quad (6)$$

In this case, the Res2Net component's output is $g = \{g_1, g_2, g_3, \dots, g_q\}$. Next, the channel size of the Res2Net residual component is confirmed by fixing it.

Enhancing appropriate fields is the goal of the Res2Net block, which improves the framework's ability to express several scales. Especially for brief utterances, this method performs better in speaker confirmation. For each feature map constructed from the feature maps, a subset of sizes was identified by the letters. x_l, where $l \in \{1,2, \dots, s\}$.

Except for x_1, all x_l are run through the 3×3 convolution filter $K_l()$. This is how the outcome of y can be expressed:

$$y_l = \begin{cases} x_l, & l = 1 \\ K_l(x_l), & l > 2 \\ K_l(x_l + y_{l-1}), & l > 2 \end{cases} \quad (7)$$

This cube moves across a convolution block that is 1×1. The procedure is stated as

$$y = Conv_{1\times1}(I) \quad (8)$$

where $y \in \mathbb{R}^{W\times H\times N}$, the number of feature mappings in the result is N. The symbol ni represents these N map characteristics with w channels, where $i \in 1,2, \dots, s$. After utilising a multi-scale approach to determine the features. The process consists of three steps:

For i=1, the convolutional process is dropped to reuse features and remove variables. The output of e_1 is displayed as

$$e_1 = n_1 \quad (9)$$

$$e_2 = Conv_{3\times3}(n_2) \quad (10)$$

Before the outcome e_{i-1} is sent to the convolutional block, it is added to the subset n_i when $s \geq i > 2$. The procedure is described as

$$e_i = Conv_{3\times3}(n_i + e_{i-1}) \quad (11)$$

This technique improves the extraction of important picture attributes that are essential for precise illness identification, like colour, shape, and texture. The algorithm can identify minor attributes in plant leaf pictures that could indicate early disease symptoms or anomalies by utilising Res2Net. This ability improves with timely and focused agricultural solutions by improving accuracy and dependability in disease identification and categorisation tasks. Furthermore, Res2Net's capacity to manage multi-scale features increases its resilience in a range of environmental circumstances and plant species, making it an essential tool for agricultural productivity and sustainable crop management.

2.4 Classification

In agriculture, disease control and early identification depend on the classification of plant leaf diseases. This study uses a hybrid methodology that combines the IDRCNN and the Bi-ConvLSTM, two effective DL techniques. Given that the former is more effective at capturing spatial information, the IDRCNN and Bi-ConvLSTM function well for this classification task.

2.4.1 Improved Deep Residual Convolutional Neural Network (IDRCNN)

IDRCNN, a version of the traditional ResNet architecture, reduces residual learning mistakes for improved feature extraction. Deep residual blocks with skip connections enable the network to learn more robust representations of leaf images, hence mitigating the vanishing gradient problem. CNN depth affects the performance of the network model. Several CNN network topologies, such as VGGNet and AlexNet, extend the network architecture to the greatest extent feasible to increase performance. Rather than imparting the underlying mapping to the stacked convolution layer, ResNet builds the stacked network to learn the residual mapping directly.

With the input x, a stacked CNN learns a feature called H(x). F(x) = H(x) − x is another map-residual mapping that it learns using stacked nonlinear layers. The output of the stacked convolutional layer receives the input immediately via the unique mapping [26].

For each convolutional layer, K grows from one to two. That means that for every size, there are 4k kernels. The remaining blocks subdivide their feedback using the appropriate shortcut link when they employ Max Pooling and the identical factor subsampling. BN and a RELU are applied before each convolutional layer. Dropout is used in between the convolutional layers and after the nonlinear layer. Ultimately, the entire SoftMax function and related layer produce five distinct kinds of ECG signal segment classification results.

2.4.2 Bi-Directional Convolutional LSTM

The Bi-ConvLSTM approach capitalises on the sequential nature of plant leaf images by employing bidirectional LSTM layers to capture temporal dependencies. Convolutional layers extract spatial characteristics from the input, while bidirectional LSTM layers model sequential patterns in the data. A BConvLS layer is currently receiving the BN phase's output. The main shortcoming of the LSTM model is that these networks need more spatial correlation as these models depend on complete relations of state-to-state and state transitions [27]. ConvLSTM proposes transforming operations into input-to-state and state-to-state transformations to tackle this issue. It consists of an input gate, an output gate ot, a forgotten gate ft, and a memory cell Ft. The memory cell access, upgrading, and cleaning are monitored by gates acting as input, output, and forget gates. What follows makes sense for ConvLSTM.

The use of convolution operations is made. Like an LSTM, the ConvLSTM contains input, output, forget, and memory gates. ConvLSTM has the following formulation:

$$i_t = \sigma\left(Z_{x_i} * X_t + Z_{h_t} * D_{t-1} + Z_{c_i} * F_{t-1} + b_i\right) \tag{12}$$

$$f_t = \sigma\left(Z_{x_f} * X_t + Z_{h_f} * D_{t-1} + Z_{c_f} * F_{t-1} + b_f\right) \tag{13}$$

$$F_t = f_t o F_{t-1} + i_t \tanh\left(Z_{x_c} * X_t + Z_{h_c} * D_{t-1} + b_c\right) \tag{14}$$

$$O_t = \sigma\left(Z_{x_o} * X_t + Z_{h_o} * D_{t-1} + Z_{c_o} o F_t + b_c\right) \tag{15}$$

$$D_t = o_t o tanh(F_t) \tag{16}$$

Here, X_t stands for the input tensor, F_t for the memory gate tensor, F_t for the hidden state tensor, and all b^*Or the bias terms.

The BConvLSTM analyses input data using both forward and backward ConvLSTMs, unlike ConvLSTM, which only employs forward data dependencies. The accuracy of predictions can be improved by the trust in the forward and backward information. The result of BConvLSTM, in the end, is

$$Y_t = tanh\left(Z_y^H \cdot \vec{D} + Z_y^H \cdot \overleftarrow{D} + b\right) \tag{17}$$

If the bias term is b, the hidden gate tensors of the forward and backward states are represented by the letters. D and $\overleftarrow{D}$, respectively, and the nonlinear function used to combine the output is called tanh. To integrate both models' strengths, combine their output layers or apply an ensemble approach. Finish the categorisation by applying a dense layer.

2.5 Segmentation

Image segmentation is a technique that attempts to separate the areas with lesions from those without by locating the region of interest (ROI) in a picture. It is simpler to discern between sections of healthy and sick leaves in the segmented image. In a grayscale image, a watershed is a transition. The shape of the image is used to split regions in watershed segmentation. The watershed uses regional minima as seeds to separate the areas. Growing methodologies based on boundaries and regions are combined in a hybrid strategy. The products can be cultivated by morphological watershed transformation after they have been labelled.

On the other hand, the watershed is a conventional technique for segmenting objects inside an image. Pixel data is treated as a local topography by the elevation approach of the watershed technique. Since the value of every picture element indicates the height of the picture at that specific location. The concepts of pixel and voxel are combined in the research under the name element. Rather than using the original image for watershed segmentation, the picture derived from the distance transform is frequently utilised.

2.5.1 EWS Algorithm

(1) Morphological restoration filter

Even while the standard smoothing filter significantly decreases noise and irregular characteristics, it changes the region contour by deleting contour edge data. Furthermore, it leaves the reconstructed image's shape unchanged. Restoring morphology is referred to as

$$M_{n+1} = (M_n \oplus StEl) \cap x \tag{18}$$

The first equation is repeated as much as is practical. When surface features and incredible disturbances are less critical than underlying components, morphological closure reconstruction and morphological restoration can recover the goal edge. On the other hand, if morphological or closed reclamation is all that is used, then one disturbance or detail can be removed from the picture, which will cause the target shape to move [28]. It is feasible to simultaneously eliminate the surface subtleties and the hiding loudness by

using the crossbreed introduction and last rebuilding methods. When reclamation actions with partial starts and completions are employed,

$$H_{StEl}^{rs} = I_{StEl}^{rs}[F_{StEl}^{rs}(x), x] \tag{19}$$

The first action, restoration, is denoted by $I_{StEl}^{rs}(x)$, and FStElrsx denotes the final restoration.

(2) Markup extraction

Only the objective area's minimal value may be maintained to avoid over-segmentation issues. $T_h$is fixed, however, and the adaptation is solitary. The adaptive acquisition approach can be used to select the threshold. T_h to prevent artificial setting parameters. The two primary difficulties with this strategy are figuring out the T_h (threshold) and locating the lowest value whose depth is less than T_h. One advantage of the T_h-minima approach is that the threshold can be obtained instantly.

T_h is automatically generated by applying the best intergroup variance approach to identify the objective region exhibiting plant disease.

Step 1: As T_h is a threshold, it separates the objective image into two groups: pixels with an area of {0,10,0}. The goal group G_0 is made up of T_h, and its picture is made up of pixels from T_h+1, T_h+2,0, and L-1.

Step 2: Compute the goal group's G_0 intragroup variation and the background group's G_0 intergroup variation. The group varies in the following ways:

$$Var^2(T) = T_0 Var^2(0) + T_1 Var^2(1) = \sum_{a=0}^{T_h}[a - \overline{U_0}]^2 T_a + \sum_{a=T_h+1}^{L-1}[a - \overline{U_1}]^2 T_a \tag{20}$$

The goal group G_0 has a mean value of $\overline{U_0}$, whereas the background group G_1's is $\overline{U_1}$. $Var^2(0)$ and $Var^2(1)$, respectively.
The variations among the classes are

$$Var^2(n) = T_0(\overline{U_1} - \overline{U_T})^2 + T_1(\overline{U_1} - \overline{U_T})^2 = T_0 T_1(\overline{U_1} - \overline{U_0})^2 \tag{21}$$

The entire variance is

$$Var^2(A) = Var^2(n) + Var^2(T) \tag{22}$$

Step 3: Decide on the best point.

$$A_h = arg \min_{0 \le T_h < L} \{Var^2(T)\} \tag{23}$$

or

$$A_h = arg \max_{0 \le T_h < L} \{Var^2(n)\} \tag{24}$$

(3) Mark-Based watershed transformation

The T_h is generated by applying the technique. The marker is then forced to appear at any rate value by using the extended change approach to the gradient-recreated picture. H_{StEl}^{rs}. Using the gradient picture H^{mark}, the minimal value forced minimum approach is used for watershed segmentation.

$$H_{EWS} = EWS(H^{mark}) \tag{25}$$

The EWS() algorithm displays the transformation when the value is H_{EWS}.
ROI identification makes it feasible to extract features from the relevant and targeted region of the leaf image.

3. Result and discussion

Employing the benchmark plant leaf disease dataset, this part investigates the detection performance of the proposed hybrid model for plant diseases. Multiple measures are applied to the experimental results. The installation details and analysis of the results are given in the ensuing sections.

3.1 Hyper-Parameter settings

In hyper-parameter tuning, iterative work may be necessary to achieve optimal performance; thus, pay attention to the training process and make the required modifications. Setting the hyper-parameters is shown in Table 1.

3.2 Experimental setup

Every model employed in this study included GPU support. For all experimental investigations, 8 GB of RAM, an Intel (R) Xeon (R) Gold CPU running at 2.20GHz. The open-source Keras 2.3.1 deep neural network framework, written in Python, implements all scripts.

Table 1 Hyper-parameter specifications.

Hyper-parameters	Epochs
Epochs	100
Dropout	0.5
Regularization	Batch Normalization
Activation	ReLu
Output Classes	38
Learning Rate	0.001

3.3 Dataset description

Images of plant leaf diseases are gathered for training and testing using the publicly available PlantVillage dataset [29]. Thirty-eight database classes contain images of healthy and damaged leaves from 14 different plant species. All of the pictures were taken in a laboratory. Figure 3 displays several sample pictures of leaves from the PlantVillage databases. The leaf picture sets are split 80-20 for train and test to evaluate assessment.

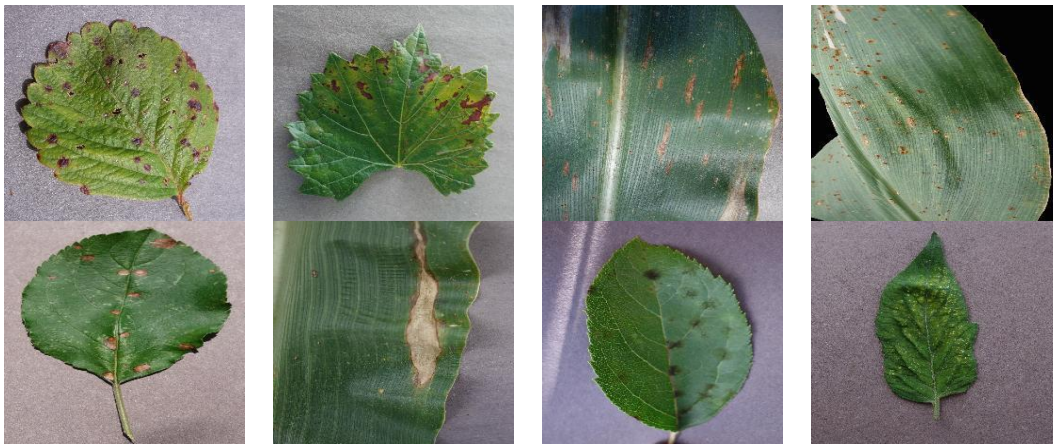


Figure 3 Example picture from some categories from the dataset.

3.4 Quantitative evaluation

Utilizing the hybrid DL model to identify leaf diseases is necessary for evaluating the proposed approach. Table 2 illustrates how well the proposed approach categorizes various plant leaf diseases.

3.5 Evaluation metrics

Consider the following performance indicators to evaluate the model's effectiveness: recall, f1-score, precision, accuracy, and the amount of parameters.

Accuracy
The formula for classification accuracy is the amount of correct forecasts divided by the total amount of accurate guesses.

$$Accuracy = \frac{Number\ of\ Correct\ Predictions}{Total\ Number\ of\ Predictions} \tag{26}$$

Precision
A helpful indicator to evaluate the model's efficacy is often classification accuracy. For example, when classes are distributed unevenly, this occurs. Nevertheless, assuming that every sample is of the best grade is plausible. Precision is the difference find when measuring the same part again using the same tools. Precision is one of these measures, and it has the following attributes:

$$Precision = \frac{TP}{(TP+FP)} \tag{27}$$

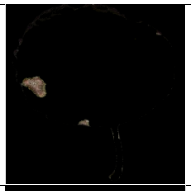
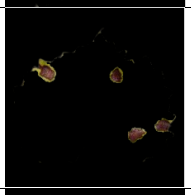
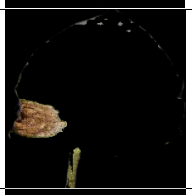
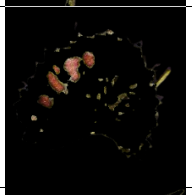
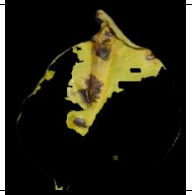
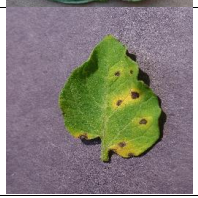
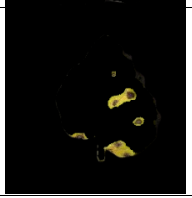
Recall
Another crucial statistic is recall or the percentage of input the system accurately predicts. This is ascertained by applying the subsequent formula:

$$Recall = \frac{TP}{(TP+FN)} \tag{28}$$

F1 Score
The F1 score is a recognized metric that combines recall and precision, and it is defined as follows:

$$F1 - score = 2 * \frac{Precision*Recall}{(Precision+Recall)} \tag{29}$$

Table 2 Classifying plant leaf diseases produced a result that showed the technique's effectiveness.

Input Picture	Pre-processed Outcome	Categorization	Segmentation Outcome of Threshold	Lesion Segmentation
		Apple_Black_rot		
		Apple_Apple_scab		
		Grape_Black_rot		
		potato_Late_blight		
		Tomato_Late_blight		
		Grape_Esca		
		Peach_Bacterial_spot		
		Pepper_bell_Bacterial_spot		
		Tomato_Early_Blight		
		Tomato_Septoria_leaf_spot		

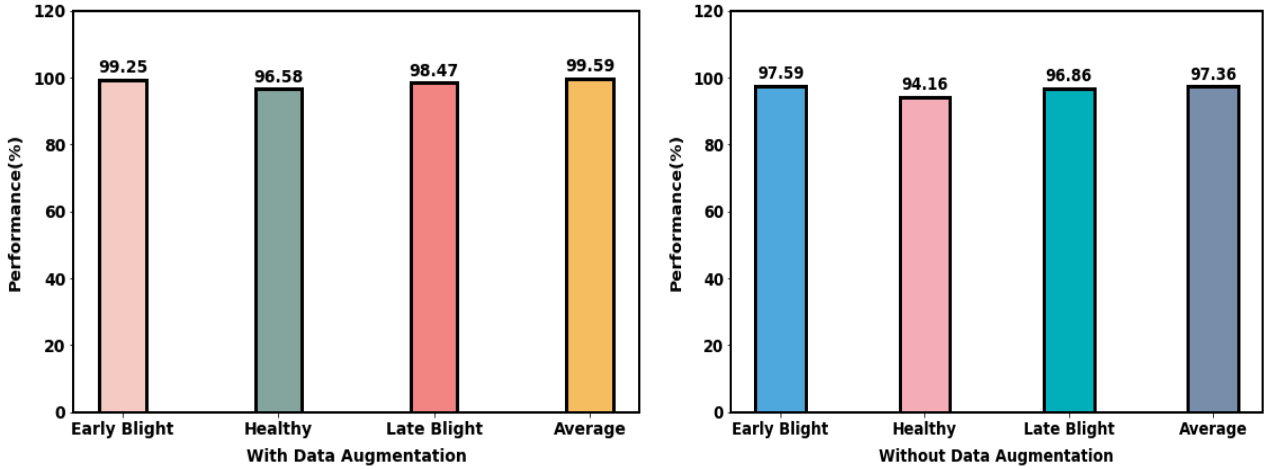
3.6 Performance valuation

The trial results showed that the proposed technique for early, healthy, and late blight correspondingly had accuracy rates of 99.25%, 98.47%, and 96.58%.

Table 3 Precision, f1-score, recall, and categorization accuracy of the proposed method.

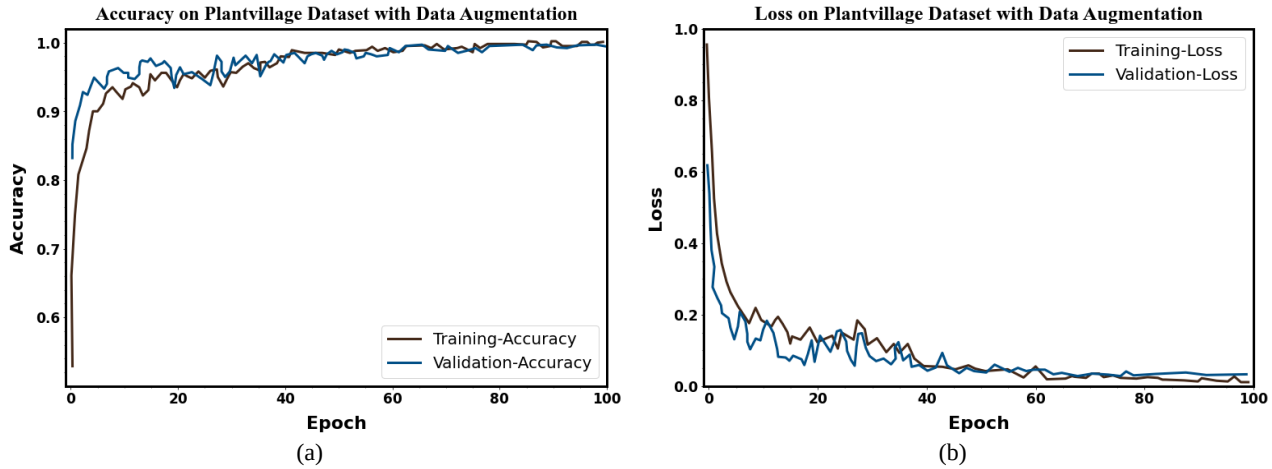
	Measures	Early Blight	Late Blight	Healthy	Average
With MCGAN	Recall	99.27%	97.95%	96.42%	99.34%
	Precision	97.13%	98.54%	100%	99.52%
	Accuracy	99.25%	98.47%	96.58%	99.59%
	F1-score	98.19%	98.24%	98.18%	99.43%
Without MCGAN	Precision	95.52%	96.57%	98.35%	96.45%
	Recall	97.59%	96.86%	94.16%	97.36%
	Accuracy	97.59%	96.86%	94.16%	97.36%
	F1-score	96.84%	95.41%	96.85%	96.09%

The Plant Village dataset (Table 3 and Figure 4) produced an accuracy of 99.59% on average when the proposed model was applied. The validation and training accuracy and losses are shown in Figure 5 for each time. Using data augmentation approaches to employ the training set from the Plant Village dataset, the results demonstrated that the proposed method successfully generalized the proposed model.



(a) (b)

Figure 4 The accuracy graph of the proposed model was obtained using two datasets: (a) with and (b) without augmentation.

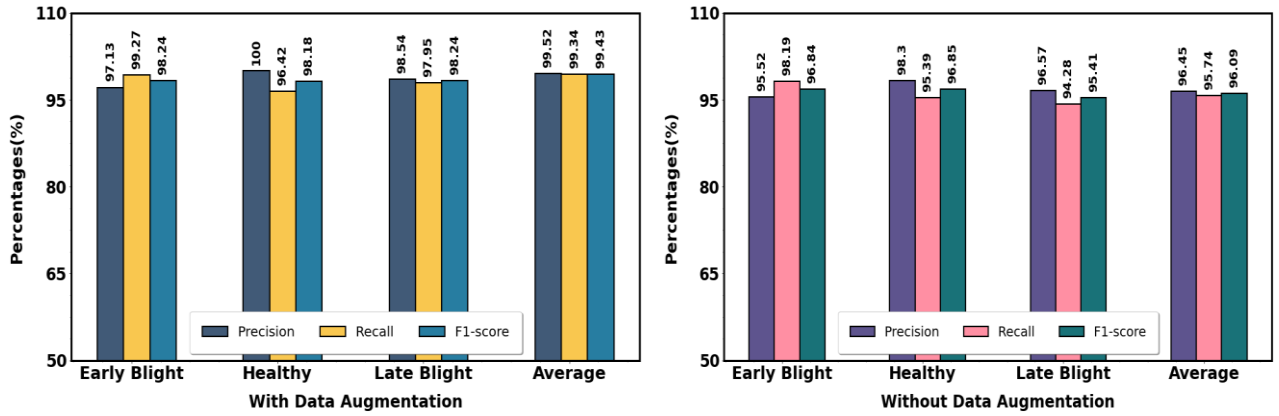


(a) (b)

Figure 5 (a) The proposed approach's accuracy; (b) the loss on the dataset using augmentation.

Additionally, it was observed that, as shown in Table 3 and Figure 6, the performance of the proposed approach yielded excellent results when data augmentation approaches were applied to the dataset training set.

Recall for the early, late, and healthy blight classes was 96.42%, 99.27%, and 97.95%, respectively. It received 98.18%, 98.19%, and 98.24% for healthy, early blight, and late blight F1 scores, respectively. The outcomes demonstrated that all of the classes in the PlantVillage dataset performed remarkably well when data augmentation techniques were applied.



(a) (b)

Figure 6 The proposed model's recall, precision, and f1-score on the datasets (a) and (b) with and without augmentation.

3.7 Comparison with base approaches

The proposed method's robustness has been tested against many DL-based methods, including GoogleNet, ResNet-101, SE-ResNet50, VGG-19, and Xception. The obtained outcomes are demonstrated in Table 4. As Table 4's data, the proposed approach achieved better performance than the baseline models.

Table 4 A comparison of the fundamental techniques and the proposed approach.

Model	Accuracy (%)	Recall (%)	Precision (%)	F1-score (%)
Xception	88.16	88.14	88.25	88.19
SE-ResNet50	96.81	96.81	96.77	96.79
ResNet-101	90.13	90.13	89.95	90.04
VGG-19	90.42	90.47	90.39	92.43
GoogleNet	87.27	87.09	87.16	87.12
Proposed Model	99.59	99.34	99.52	99.43

Regarding precision, recall, F1-score, and accuracy, the GoogleNet approach produces the lowest average scores—87.12%, 87.27%, 87.09%, and 87.16%, respectively. However, the average precision, F1-score, recall, and accuracy values for the Xception model are 88.14%, 88.16%, 88.19%, and 88.25%, respectively, achieving the second minimum values. In contrast, the proposed strategy yields the most significant results, with average precision, F1-score, recall, and accuracy values of 99.52%, 99.43%, 99.34%, and 99.59%, respectively. The proposed strategy produces an average memory value of 99.34%, compared to the comparative approaches' 90.53% average recall value. Furthermore, the average accuracy and f1-Score for the basic models are 90.56% and 90.91%, respectively, whereas the average values for the proposed method are 99.59% and 99.43%.

The data indicates that the proposed approach performs better at classifying and diagnosing leaf diseases in tomato plants. The comparative study between the base and proposed approaches is displayed in Figure 7.

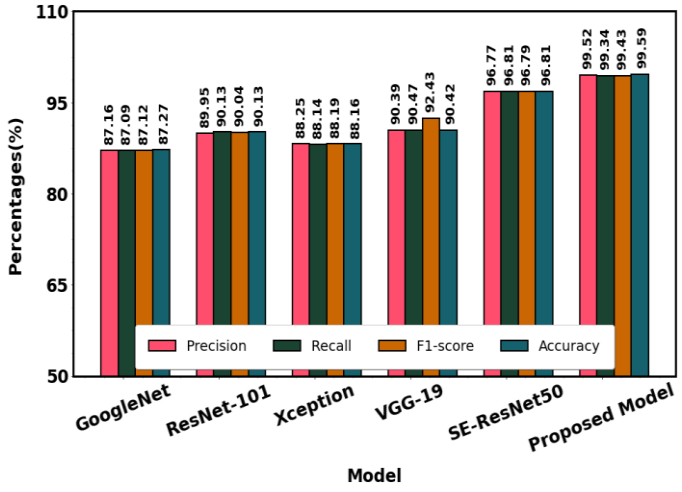


Figure 7 A comparison between the proposed approach and the foundational methods.

3.8 Comparison with existing literature

To verify the proposed model's claims, compare its performance with other models already in the literature. Table 5 presents a comparison of model-based performance. Table 5 presents a comprehensive assessment of the proposed system. Additionally, Table 5

demonstrates that in contrast to other models. Additionally, Table 5 indicates that the models were used on the PlantVillage dataset, which consists of images captured in a controlled environment.

Table 5 Comparing this plant disease classification study with others.

Reference	Techniques	Average Accuracy
Abbas et al. [16]	DenseNet121	97.11%
Ashwinkumar et al. [17]	OMNCNN	98.7%
Elaraby et al. [18]	AlexNet	98.83%
Umamageswari et al. [19]	PNAS	97.43%
Syed-Ab-Rahman et al. [20]	Faster-RCNN	94.37%
Nalini et al. [21]	DNN-CSA	96.96%
Proposed Approach	IDRCNN-BDCLSTM	99.59%

The proposed approach produced higher accuracy when compared to other existing techniques, and the proposed model performs well. The overall comparison between the proposed system and previous approaches in the literature is shown in Figure 8. A confusion matrix table can be used to visualize how well a DL network diagnoses illnesses on plant leaves. The matrix compares the actual ground truth and the predicted outcomes.

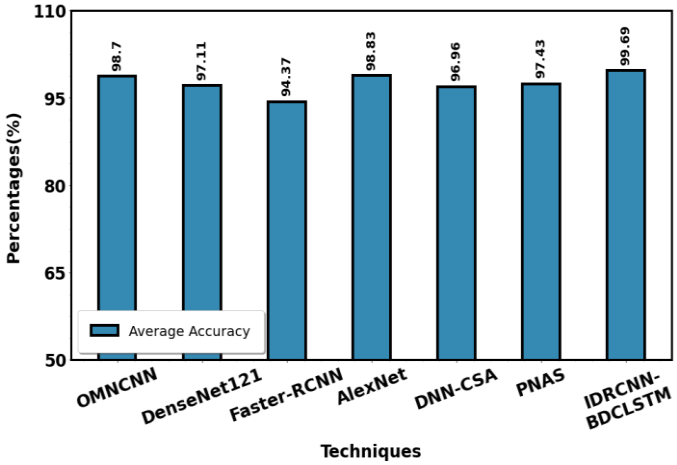


Figure 8 Comprehensive comparison of proposed and existing techniques.

The differences between the model's predicted and actual results are deeply broken down in a confusion matrix. Figure 9 illustrates the confusion matrix used to categorize plant leaf diseases.

3.9 Training and testing evaluation

During the training phase, training loss and accuracy serve as the primary learning criteria for the system. They facilitate the assessment of the model's degree of integration with the training data. On the other hand, testing accuracy and loss is what counts for generalization assessment and model evaluation. They are concerned with the system's anticipated assessment of novel, untested data. Figures display the accuracy in testing and training and the loss functions. Because of the training stage, the proposed approach is trained using the given training set for 100 epochs. It has been determined that the learning rate is 0.1. The training evaluation and testing simulations of accuracy and loss are shown in Figure 10.

3.10 Computational time

The elements that impact the running time cost of leaf illness categorization include the dataset's complexity and size and the DL approaches used. Analyzing imaging data is typically necessary to classify patients into various disease categories appropriately. The computation times of the proposed and existing methods are contrasted in Table 6.

Table 6 Comparing the computation times of proposed and previous approaches.

References	Approaches	Running time (ms)
Abbas et al. [16]	DenseNet121	0.32
Ashwinkumar et al. [17]	OMNCNN	0.25
Elaraby et al. [18]	AlexNet	0.23
Umamageswari et al. [19]	PNAS	0.24
Syed-Ab-Rahman et al. [20]	Faster-RCNN	0.19
Nalini et al. [21]	DNN-CSA	0.36
Proposed Method	IDRCNN-BDCLSTM	0.14

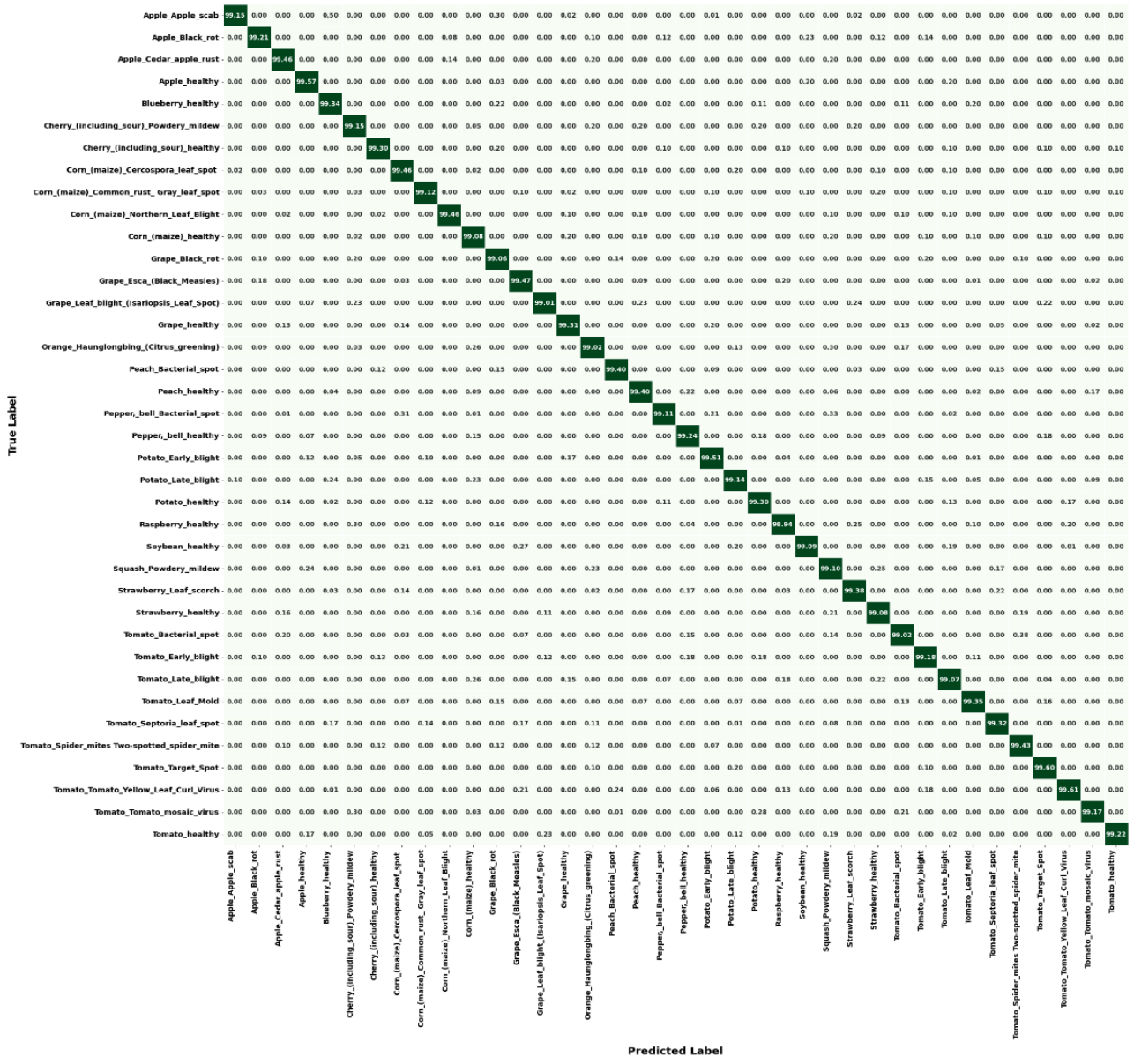
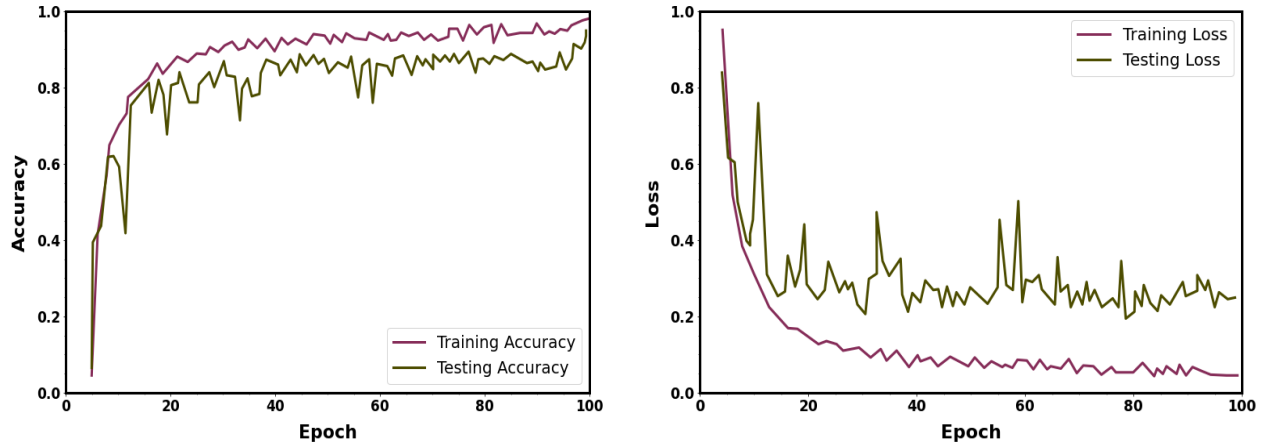


Figure 9 Confusion matrix for classification of plant leaf diseases.



(a) (b)

Figure 10 Evaluation performance of proposed approach's testing and training, (a) Accuracy, (b) Loss.

The proposed hybrid deep learning techniques produced better results with shorter computation times. It uses effective techniques and performances to lessen the difficulty of identifying diseases in plant leaves. The computational time required for the proposed approach is shown in Figure 11 compared to existing techniques.

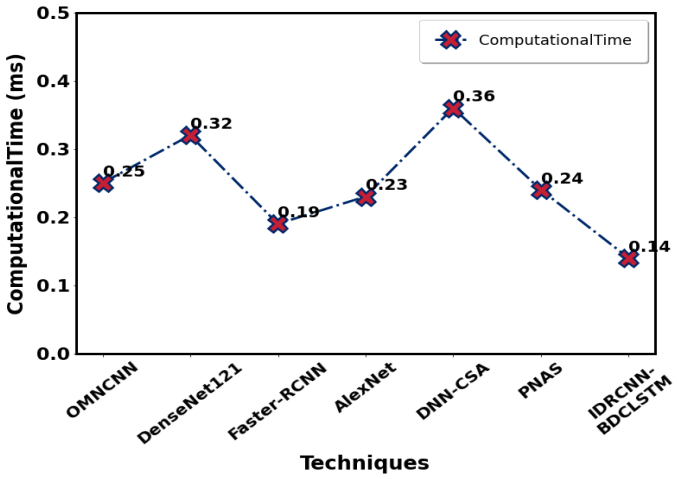


Figure 11 Time complexity of computation for both proposed and previous approaches.

4. Conclusion

Developing machine learning or deep learning models for disease identification is one method of classifying illnesses that impact plant leaves. This study eliminated noise from the input image, and data augmentation was used to lessen overfitting problems. Subsequently, they improved the image's contrast to identify the disorders precisely, and then they improved the image's background for more accurate categorization. Next, the Res2Net approach extracts the pertinent features to classify the disease. To classify it in their category, they used a hybrid DL-based approach. Finally, the EWS algorithm separates the lesion and threshold disease areas from the picture. The proposed method outperformed earlier methods regarding classification accuracy while requiring less computing time. The hybrid approaches that have been proposed perform better at classifying plant leaf diseases and minimize evaluation losses.

4.1 Limitations of the proposed model and future scopes

The proposed model, while effective, has certain limitations that warrant attention. Its reliance on high-quality, preprocessed images may limit applicability in real-world scenarios where such preprocessing isn't feasible or images are of lower quality. Additionally, the synthetic data generated through MCGAN for augmentation might not fully capture the complexity of real-world plant diseases, potentially impacting the model's accuracy in novel conditions. The computational complexity of the hybrid deep learning approach (BDC-LSTM and IDRCNN) also poses challenges, particularly in resource-constrained environments. Future research could address these limitations by developing more robust preprocessing techniques, incorporating diverse datasets to enhance generalization, and optimizing computational efficiency for broader accessibility. Moreover, integrating advanced interpretability methods and exploring real-time detection capabilities through edge computing or mobile platforms could significantly enhance the model's practical utility, making it more reliable and applicable in diverse agricultural settings.

5. Acknowledgements

We declare that this manuscript is original, has not been published before and is not currently being considered for publication elsewhere.

6. References

[1] Harakannanavar SS, Rudagi JM, Puranikmath VI, Siddiqua A, Pramodhini R. Plant leaf disease detection using computer vision and machine learning algorithms. Glob Transit Proc. 2022;3(1):305-10.
[2] Vallabhajosyula S, Sistla V, Kolli VKK. Transfer learning-based deep ensemble neural network for plant leaf disease detection. J Plant Dis Prot. 2022;129(3):545-58.
[3] Sai Reddy B, Neeraja S. Plant leaf disease classification and damage detection system using deep learning models. Multimed Tools Appl. 2022;81(17):24021-40.
[4] Sahu SK, Pandey M. An optimal hybrid multiclass SVM for plant leaf disease detection using spatial Fuzzy C-Means model. Expert Syst Appl. 2023;214:118989.
[5] Sharma V, Tripathi AK, Mittal H. DLMC-Net: Deeper lightweight multi-class classification model for plant leaf disease detection. Ecol Inform. 2023;75(1):102025.
[6] Pal A, Kumar V. AgriDet: Plant leaf disease severity classification using agriculture detection framework. Eng Appl Artif Intell. 2023;119:105754.
[7] Fan X, Luo P, Mu Y, Zhou R, Tjahjadi T, Ren Y. Leaf image-based plant disease identification using transfer learning and feature fusion. Comput Electron Agric. 2022;196:106892.
[8] Gupta S, Tripathi AK. Fruit and vegetable disease detection and classification: recent trends, challenges, and future opportunities. Eng Appl Artif Intell. 2024;133:108260.
[9] Seetharaman K, Mahendran T. Leaf disease detection in the banana plant using gabor extraction and region-based convolution neural network (RCNN). J Inst Eng India Ser A. 2022;103(2):501-7.
[10] Sharma V, Tripathi AK, Mittal H. Technological revolutions in smart farming: current trends, challenges & future directions. Comput Electron Agric. 2022;201:107217.

[11] Abd Algani YM, Caro OJM, Bravo LMR, Kaur C, Al Ansari MS, Bala BK. Leaf disease identification and classification using optimized deep learning. Meas: Sens. 2023;25:100643.
[12] Yogeshwari M, Thailambal, G. Automatic feature extraction and plant leaf disease detection using GLCM features and convolutional neural networks. Mater Today: Proc. 2023;81:530-6.
[13] Gajjar R, Gajjar N, Thakor VJ, Patel NP, Ruparelia S. Real-time detection and identification of plant leaf diseases using convolutional neural networks on an embedded platform. Vis Comput. 2022;38:2923-38.
[14] Moussafir M, Chaibi H, Saadane R, Chehri A, El Rharras A, Jeon G. Design of efficient techniques for tomato leaf disease detection using genetic algorithm-based and deep neural networks. Plant Soil. 2022;479:251-66.
[15] Kavitha Lakshmi R, Savarimuthu N. DPD-DS for plant disease detection based on instance segmentation. J Ambient Intell Human Comput. 2023;14(4):3145-55.
[16] Abbas A, Jain S, Gour M, Vankudothu S. Tomato plant disease detection using transfer learning with C-GAN synthetic images. Comput Electron Agric. 2021;187:106279.
[17] Ashwinkumar S, Rajagopal S, Manimaran V, Jegajothi B. Automated plant leaf disease detection and classification using optimal MobileNet based convolutional neural networks. Mater Today: Proc. 2022;51:480-7.
[18] Elaraby A, Hamdy W, Alruwaili M. Optimization of deep learning model for plant disease detection using particle swarm optimizer. Comput Mater Contin. 2022;71(2):4019-31.
[19] Umamageswari A, Bharathiraja N, Irene DS. A novel fuzzy C-means based chameleon swarm algorithm for segmentation and progressive neural architecture search for plant disease classification. ICT Express. 2023;9(2):160-7.
[20] Syed-Ab-Rahman SF, Hesamian MH, Prasad M. Citrus disease detection and classification using end-to-end anchor-based deep learning model. Appl Intell. 2022;52(1):927-38.
[21] Nalini S, Krishnaraj N, Jayasankar T, Vinothkumar K, Britto ASF, Subramaniam K, et al. Paddy leaf disease detection using an optimized deep neural network. Comput Mater Contin. 2021;68(1):1117-28.
[22] Sharma V, Tripathi AK, Daga P, Nidhi M, Mittal H. ClGanNet: A novel method for maize leaf disease identification using ClGan and deep CNN. Signal Process Image Commun. 2024;120:117074.
[23] Sharma V, Tripathi AK, Mittal H, Nkenyereye L. SoyaTrans: A novel transformer model for fine-grained visual classification of soybean leaf disease diagnosis. Expert Syst Appl. 2025;260:125385.
[24] Choi J, Noh KJ, Cho SW, Nam SH, Owais M, Park KR. Modified conditional generative adversarial network-based optical blur restoration for finger-vein recognition. IEEE Access. 2020;8:16281-301.
[25] Das A, Chandran S. Transfer learning with res2net for remote sensing scene classification. The 11th International Conference on Cloud Computing, Data Science & Engineering (Confluence); 2021 Jan 28-29; Noida, India. USA: IEEE; 2021. p. 796-801.
[26] Arun Pandian J, Kanchanadevi K, Rajalakshmi NR, Arulkumaran G. An improved deep residual convolutional neural network for plant leaf disease detection. Comput Intell Neurosci. 2022;2022(1):1-9.
[27] Azad R, Asadi-Aghbolaghi M, Fathy M, Escalera S. Bi-directional ConvLSTM U-Net with densely connected convolutions. 2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW); 2019 Oct 27-28; Seoul, Korea. USA: IEEE; 2019. p. 406-15.
[28] Sharma AK, Nandal A, Dhaka A, Koundal D, Bogatinoska DC, Alyami H. Enhanced watershed segmentation algorithm-based modified ResNet50 model for brain tumor detection. BioMed Res Int. 2022;2022(1):7348344,. Retraction in: BioMed Res Int. 2023;2023(1):9865972.
[29] Kaggle. PlantVillage dataset: Dataset of diseased plant leaf images and corresponding labels [Internet]. 2019 [cited 2024 Jan 27]. Available from: https://www.kaggle.com/datasets/emmarex/plantdisease.