

Improved Agricultural Monitoring through Automated Detection and Classification of Purple Blotch and Yellow Blotch Diseases on Onion Leaves utilizing Dual Branch Graph Neural Network

Mr. Bhausaheb B. Vikhe¹, Dr. Purushottam R. Patil²

^{1,2}School of Computer Sciences and Engineering, Sandip University, Nashik, Maharashtra, India.
Email: ¹bhau770.vikhepatil@gmail.com, ²purushottam.patil@sandipuniversity.edu.in

Abstract: Traditional visual inspection is replaced by plant disease detection (PDD), which integrates plant pathology, image processing, remote sensing, agronomy and machine learning. Remote sensing tools and high-resolution drone imagery enhance real-time disease monitoring. However, challenges like data scarcity and algorithm generalization make accurate and automated disease diagnosis more complex. In this manuscript, Improved Agricultural Monitoring through Automated Detection and Classification of Purple Blotch and Yellow Blotch Diseases on Onion Leaves utilizing Dual Branch Graph Neural Network (ADC-PBYBD-OL-DBGNN) is proposed. The onion leaf dataset was first collected in the Karnataka village of Chilwadigi, which has a variety of climates. Then the collected images are preprocessed using Distributed Adaptive Spatial Filtering (DASF) is used for resizing and cropping. The preprocessed images are fed to Onion Leaf Disease Detection utilizing Dual Branch Graph Neural Network (DBGNN) for detecting and classifying as Healthy, Iris Yellow Virus, Purple Blotch and Leaf Blight. DBGNN generally doesn't show any adaptation of optimization techniques for figuring out the best parameters to guarantee precise onion leaf detection. Hence, Superb Fairy-wren Optimization Algorithm (SFOA) is utilized to optimize DGRGNN for precisely classifying the onion leaves. Then the proposed ADC-PBYBD-OL-DBGNN is implemented and the performance metrics like Accuracy, Precision, Recall, F1 score, Computational time are analyzed. Finally, the performance of proposed ADC-PBYBD-OL-DBGNN method provides 26.68%, 25.75%, and 26.16% higher accuracy and 27.49%, 24.75%, and 25.85% higher precision while compared with existing methods such as the onion plant leaf image dataset for classification and detection (OPLM-CD-Xpection), the onion and maize image datasets for creating AI-based classification models for pests and diseases (OMI-PD-CNN) and the meta-learning based neural architecture search for automated generation of deep neural networks for plant disease detection tasks (SAG-PDD-DNN) correspondingly.

Keywords: Distributed Adaptive Spatial Filtering, Dual Branch Graph Neural Network, Superb Fairy-wren Optimization Algorithm and Onion Leaf Disease.

1. Introduction

A system that analyzes and interprets visual data in real time is called computer vision, and it can partially mimic human ability [1–3]. Computer systems can effectively solve complex problems and attain higher computational intelligence by analyzing real-world data captured by cameras and modeling human vision. Gathering images, processing them, extracting features, and identifying objects are the basic steps of digital image processing that help us understand visual data. Computer vision is used in many different fields, including smart agriculture and medical diagnostics. A farming technique that employs cutting-edge technology to improve agricultural quality and yields is known as "smart agriculture," which is another term for the third green revolution [4–6]. These technologies include things like the Internet of Things, drones, cameras, sensors, and big data. Fields are home to devices that



collect data, such as sensors and cameras. Sensors and cameras installed in fields continuously collect data on environmental conditions, crop health, and soil qualities, enabling for more precise decision-making and resource optimisation [7]. Computer vision algorithms analyze this data to detect plant illnesses, track growth patterns, and check soil moisture levels, allowing for automated actions that improve agricultural efficiency [8]. One of the most popular uses of computer vision is smart agriculture, sometimes known as the third green revolution; in which modern technology improve farming efficiency and crop quality [9]. This approach monitors agricultural areas using high-resolution cameras, drones, IoT devices, and big data analytics. Farmers are able to make decisions based on data because of sensors placed on farmlands that continuously gather information on plant health, weather patterns, and soil conditions. This data is used by computer vision algorithms to identify plant diseases, assess crop growth, and maximize resource use, leading to higher yields and more environmentally friendly farming practices [10].

Remote sensing, image processing, machine learning, pathology, and agronomy are all used in plant disease detection research. Advanced technology is replacing traditional visual inspection methods. Examples of remote sensing tools that provide comprehensive views of crop health are UAVs and satellites. Drones provide high-resolution imagery for real-time disease monitoring. Image processing algorithms transform unprocessed data into meaningful information.

Automating disease diagnosis is aided by machine learning. There are still issues like algorithm generalization and data scarcity. Progress requires interdisciplinary teamwork. Early disease detection and effective management are intended to be facilitated by the solutions. The ultimate goals are to advance sustainable agriculture and safeguard global food security.

The proposed method introduces a novel DBGNN framework for automated classification and detection of Purple Blotch and Yellow Blotch diseases in onion leaves, addressing limitations in existing methods. Unlike conventional models, DBGNN effectively captures spatial and structural relationships within leaf images, enhancing classification accuracy. The integration of the SFOA optimizes DBGNN's parameters, ensuring improved precision in disease identification. This research advances the creation of more reliable and flexible disease detection methods.

Major contribution of this paper includes,

- Utilizes onion leaf images collected from Chilwadigi village, Karnataka, under varied climatic conditions, ensuring robustness in real-world agricultural monitoring.
- DASF is utilized for resizing and cropping images and reducing computational complexity.
- The SFOA to optimize DBGNN parameters, improving classification accuracy and robustness.
- Enhances disease detection accuracy by leveraging graph-based feature learning in DBGNN, which effectively captures spatial and topological relationships in leaf images.

The balance paper is arranged as below: part 2 presents literature survey, part 3 defines proposed approach, part 4 illustrates result with discussion, part 5 concludes the manuscript.

2. Literature Survey

Several investigate works were presented in literature to onion plant leaf disease detection utilizing DL; some current works are reviewed here;

In 2024, Aishwarya, M.P. and Reddy, [11] have presented dataset of images of onion plant leaves for detection and classification. Here, a comprehensive dataset of onion leaves was collected from the Karnataka village of Chilwadigi, which has a variety of climates. A variety of carefully selected onion leaf categories were included in the dataset to address the complex issues of classifying leaf images taken in natural environments, including lighting variations, subtle inter-class similarities, and background conditions such as different foliage arrangements and light levels. It offers low precision as well as high accuracy.

In 2024, Verma, S., et.al, [12] have presented Deep neural networks were automatically generated for plant disease detection tasks utilizing meta-learning-based neural architectures. Here, CNN models were automatically constructed for previously encountered plant disease detection tasks using a two-phase neural architecture search system (ML NAS) based on meta-learning. Based on previous benchmark model evaluations on current plant disease datasets, the first step provides the best benchmark models for unknown PDD tasks. The second step involves optimizing the suggested model for the intended task using the recommended NAS operators. It offers both low accuracy and high precision.

In 2025, Appiah, O., et.al, [13] have presented TOM2024: Image datasets of maize and onions for creating AI-based classification models for diseases and pests. The TOM2024 dataset was a large collection of high-quality images for identifying diseases and pests in maize and onion crops. It includes over 12,000 labeled images in 30 classes (pest, infested crop, and healthy crop) and 25,844 initial images across three cropping systems. The dataset, which encompasses a variety of environments, was obtained through painstaking fieldwork in Burkina Faso utilizing high-resolution cameras. It offers low precision along with a high F1 score.

In 2024, Saraswathi, E. and Banu, J.F., [14] have presented for the classification of plant leaf diseases; a hybrid CGAN combines OTSU and surf feature extraction. Plant leaf diseases can be categorized using the hybrid CGAN-OF model, which combines OTSU and SURF. The suggested CGAN-OF model preprocesses and enhances images using contrast-limited adaptive histogram equalization. Using scale-invariant feature transformations, the suggested model extracts local features using the SURF technique and speeds up image segmentation using the OTSU method without any prior knowledge of the images. By employing the picture generation technique, CGAN broadens the input plant village dataset and detects and categorizes a variety of plant leaf diseases. There were three categories of leaf diseases: bacterial, viral, and fungal. It has a short computation time and a low recall.

In 2025, Chai, A.Y.H., et.al, [15] have presented a Novel Baseline Model for Improved Plant Disease Identification by Combining Local Features and Global Attention. Local and Global Characteristics of Plant Disease The Fusion Attention Model (PlantAIM) was a hybrid framework that combines CNN's local feature extraction capabilities with ViT's global attention mechanisms. The goal of PlantAIM was to improve the model's capacity for learning and to concentrate on traits unique to both crops and diseases. Perform comprehensive tests to evaluate PlantAIM's generalizability and robustness against state-of-the-art (SOTA) models, taking into account situations with a lack of training data and actual environmental data. It offers both high computational time and high recall. Table 1 displays the literature survey's comparison.

Table 1: Literature Survey's Comparison

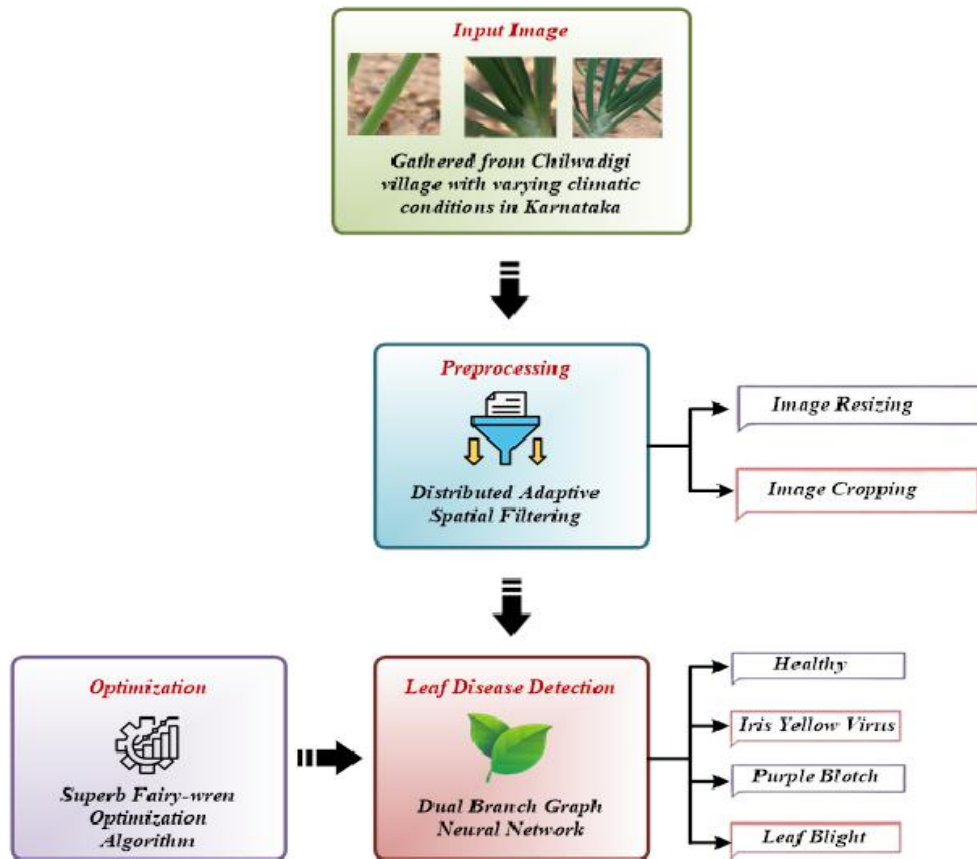
| Author (s) | Methods | Objectives | Advantages | Disadvantages |
|--|--|--|--|---|
| In 2024, Aishwarya, M.P. and Reddy, [11] | Image dataset collection | Provide a dataset of pictures of the leaves of onions and chillies for tasks involving detection and classification. | Supports plant disease classification research | No model or analysis provided |
| In 2024, Verma, S., et.al, [12] | Meta-learning-based Neural Architecture Search (MLNAS) | Automate deep neural network design for plant disease detection | Reduces manual architecture design effort | High computational cost for searching optimal architectures |
| In 2025, Appiah, O., et.al, [13] | Image dataset collection | Provide datasets (TOM2024) of onion, tomato, and maize images for AI-based pest and disease classification | Facilitates AI-based plant disease research | Lacks benchmark results for classification performance |
| In 2024, Saraswathi, E. and Banu, J.F., [14] | Hybrid Conditional GAN (CGAN), OTSU, SURF feature extraction | Improve plant leaf disease classification utilizing hybrid CGAN and feature | Enhances classification accuracy | CGAN training requires large datasets and fine-tuning |

| | | extraction methods | | |
|------------------------------------|--|--|--|---|
| In 2025, Chai, A.Y.H., et.al, [15] | Global attention and local feature integration in PlantAIM model | Enhance plant disease identification through improved feature extraction | Better feature representation for disease classification | May require high computational resources for training |

Table 1 highlights recent advancements in plant disease detection, including dataset contributions automated neural architecture design hybrid deep learning approaches and enhanced feature extraction. While these studies improve classification accuracy, challenges such as high computational costs, large dataset requirements, and the lack of standardized benchmarks remain. This manuscript addresses these issues by proposing an optimized approach to enhance classification efficiency, reduce complexity, and improve model generalizability.

3. Proposed Methodology

The proposed strategy in this field entails gathering an onion leaf dataset from Chilwadigi village in Karnataka under various weather circumstances. The images undergo preprocessing using DASF for resizing and cropping. The preprocessed images are then analyzed using DBGNN for detecting and classifying leaves as Healthy, Purple Blotch, Iris Yellow Virus, and Leaf Blight. To enhance classification accuracy, SFOA is employed to optimize DBGNN by determining optimal parameters for precise detection. Figure 1 displays the block diagram of the proposed ADC-PBYBD-OL-DBGNN approach.



3.1 Data Acquisition

The onion leaf dataset was first collected in Chilwadigi village [16]. All of the images were taken in Chilwadigi village, which is situated in the Koppal district. Throughout the Kharif season, the data is collected. The images were taken in rural locations with a Canon Mark II D digital camera. Four folders have been created from the onion leaf image dataset: "Picture labels include "healthy" (1278 images), "Iris yellow virus" (1272 images), "purple blotch" (735 images), and "Leaf Blight" (1217 images). Plant pathologists help with the labeling.

Table 2: The state of the onion leaf, as well as the total count of each state

| Category | No of Images |
|-------------------|--------------|
| Leaf Blight | 1217 |
| Iris yellow virus | 1272 |
| Purple Blotch | 735 |
| Healthy | 1278 |
| Total | 4502 |

3.2 Advanced Image Preprocessing using Distributed Adaptive Spatial Filtering

In this section, Preprocessing using (DASF) [17] is discussed. DASF is used for cropping and resizing. DASF improves preprocessing by optimizing cropping and scaling to preserve important features while deleting extraneous data. It increases computing performance, lowers memory utilization, and improves visual clarity. DASF achieves correct feature extraction by adaptively filtering spatial noise and distortions, resulting in enhanced model performance in analytical tasks. Initialize spatial filtering with kernel size, noise reduction thresholds, and adaptive grid partitioning, and set target image dimensions while maintaining aspect ratios as given in equation (1).

$$\hat{x}_k^i = Y_k^{iT} sk \quad (1)$$

Here, \hat{x}_k^i represents input onion leaf images, Y_k^{iT} denotes ensure variations in lighting and sk stands for pixel intensity. Identify and crop relevant leaf regions while preserving essential structural features. Thus, it is given in equation (2).

$$\hat{F}_{j,k} = Y_k^{iT} F_{j,k} \quad (2)$$

Here, $\hat{F}_{j,k}$ indicates the sub region in the image, $F_{j,k}$ is the adaptive filtering kernel. Maintain aspect ratio while scaling the cropped region to a predefined size, using interpolation techniques to enhance resolution and preserve texture details as given in equation (3).

$$m^{i+1} = Y^{i+1Y} x = \sum_k Y_k^{i+1Y} xk = \tilde{Y} * \tilde{x}^i \quad (3)$$

Where, m^{i+1} represents the set of pixels forming the cropped leaf region, Y^{i+1Y} denotes gradients in the horizontal and vertical conditions, x is the resized image, xk denotes pixel coordinates in the resized space, \tilde{Y} stands for set of pixels forming the resized leaf region and \tilde{x}^i denotes final output. Figure 2 displays cropped images of onion leaves.

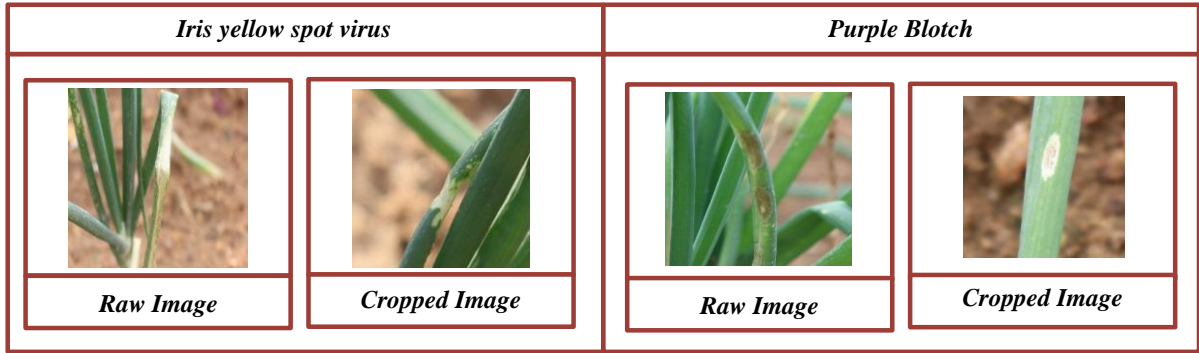


Figure 2: Examples of Cropping and Resizing of Onion Leaves

Finally, DASF has successfully resized and cropped. Then the preprocessed images are fed to detection phase.

3.3 Dual Branch Graph Neural Network for Precise Detection of Onion Leaf Diseases

In this section, Onion Leaf disease Detection using (DBGNN) [18] is discussed. DBGNN is used to detect and classify as Healthy, Iris Yellow Virus, Purple Blotch and Leaf Blight. DBGNN has various advantages, including improved classification accuracy because it effectively captures spatial and structural correlations in leaf images. The dual-branch architecture enhances feature extraction, allowing for exact distinction of diverse disease states. DBGNN is also resistant to fluctuations in lighting, angle, and background noise, making it ideal for real-world agricultural applications. Its capacity to simulate intricate connections between leaf sections improves disease localization, allowing for early identification and appropriate action for better crop management. Figure 2 depicts the architecture of DBGNN.

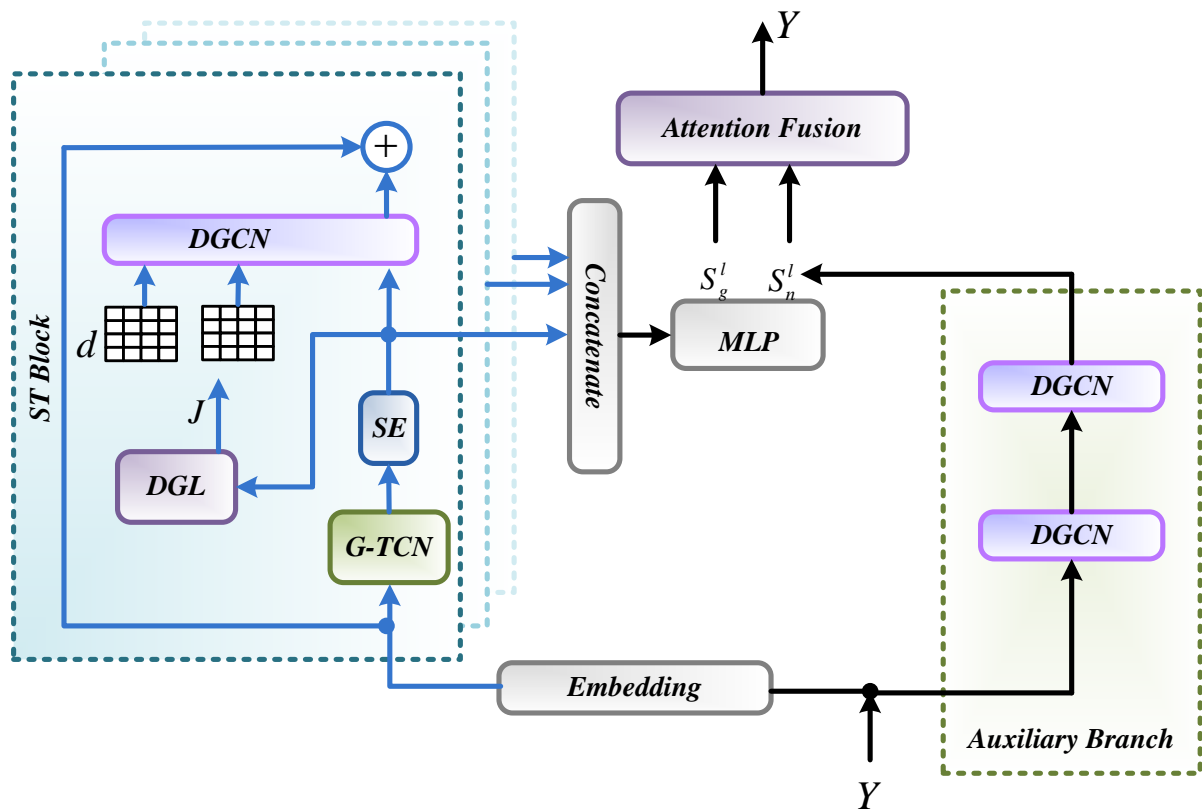


Figure 3: The overall architecture of DBGNN

A single Fusion module and two branches comprise the overall architecture. Several spatiotemporal blocks, an embedding layer, and an MLP layer make up the main branch. A Diffusion Graph Convolution (DGCN) function,

a Squeeze and Excitation (SE) function, a Gated Temporal Convolutional module (G-TCN), and a Dynamic Graph Learning (DGL) layer make up each spatiotemporal block. There are shallow levels of DGCN in the auxiliary branch. Spatial correlations in each spatiotemporal block of the main branch are indicated by combining the dynamic graph S_n^l that DGL learned with the predefined static graph S_g^l . But in the auxiliary branch's static graphs, only each DGCN uses the predefined static graph S_g^l . Graph-based models encode leaf structure and topology, with GCNs or GATs refining node embeddings to capture disease patterns as given in equation (4).

$$d * \theta(y) = \sum_{i=0}^{L_d-1} \theta(i) d(y - f \times i) \quad (4)$$

Here, $*$ indicates the convolution operation, y denotes the current time step, f is represented as the dilation factor, L_d is indicated as the size of the convolutional kernel. An attention-based mechanism fuses outputs from both branches, integrating spatial and structural features for classification. Thus, it is given in equation (5).

$$j = \tanh(\theta_1 * Y + f) \oplus \text{sigmoid}(\theta_2 * Y + v) \quad (5)$$

Here, j denotes number of disease categories, θ_1, θ_2, f and v are model factors, \oplus indicates the Hadamard product, $\tanh(\cdot)$ acts as the input's activation function and $\text{sigmoid}(\cdot)$ establishes the gating ratio for the latent representations' filtering. Equation (6) provides the classification of the fused features into categories such as healthy and diseased after they have been processed through a fully connected layer.

$$J = \tilde{S}CE \quad (6)$$

Where, J represents the output, \tilde{S} denotes number of iris yellow virus diseases, C represents the number of purple blotch disease, E is the number of leaf blight disease. Finally, DBGNN has successfully detected and classified as Healthy, Iris Yellow Virus, Purple Blotch and Leaf Blight. Here, SFOA is used to optimize DBGNN by tuning the weight parameters and bias parameters.

3.4 Optimization utilizing Superb Fairy-wren Optimization Algorithm

In this sector, Optimization using (SFOA) [19] is utilized to enhance weight parameters L_d and θ_1 of proposed DBGNN. The SFOA enhances DBGNN by improving convergence speed, preventing local optima, optimizing weight adjustments, and ensuring stable learning, leading to better accuracy and efficiency. Originally from southeast Australia, the Superb Fairy-wren is a small bird that weighs 8–13 g and measures about 14 cm. The plumage of females and chicks is yellow-brown, while that of males is blue-black and gray. These birds inhabit moist forests and practice cooperative breeding in groups of three to five, with males often assisting in chick care. They teach survival and reproductive skills through migration, and males display courtship behaviours during breeding. Their diet consists mainly of insects, along with seeds, flowers, and fruit. They use warning calls and body signals to alert their group to predators, demonstrating strong social behavior and adaptability. Here, step by step procedure for acquiring regulates DBGNN utilizing SFOA is described. Each step method is given in below.

Step 1: Initialization

In order to effectively handle real-world optimization problems, the proposed SFOA approach is a population-based strategy that alters evaluation durations to simulate its search capabilities in space. The problem article in the search space will be used by each SFOA member to ascertain the problem decision variable's value. Each SFOA member is represented mathematically by vectors and represents a possible solution to the issue. The algorithm equation (7) is generated by the combination of all SFOA members, with a decision variable represented by each vector element.

$$Y = \begin{bmatrix} Y_1 \\ \vdots \\ Y_i \\ \vdots \\ Y_M \end{bmatrix}_{M \times F} = \begin{bmatrix} y_{1,1} & \cdots & y_{1,D} & \cdots & y_{1,D} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ & \cdots & & \cdots & y_{i,D} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ y_{M,1} & \cdots & y_{M,d} & \cdots & y_{M,D} \end{bmatrix}_{M \times D} \quad (7)$$

Here, Y is denoted as the SFOA global matrix, Y_i is indicated as the i^{th} SFOA member (candidate solution), $y_{i,D}$ are represented as the D dimensions of SFOA (decision variables) in the search space and M is indicated as the number of global members.

Step 2: Random Generation

Following setup, random factors were created for the input. It is dependent upon an explicit hyper parameter requirement.

Step 3: Fitness Function

It uses initialized values to produce an arbitrary solution. It is calculated by exploration node. Then the formula is derived in equation (8).

$$Fitness\ Function = Optimizing[L_d\ and\ \theta_1] \quad (8)$$

Here, the factor L_d is employed to increase the accuracy and θ_1 is employed to decrease the Computational time.

Step 4: Young Bird Growth Stage for Optimizing L_d

During the growth phase of SFOA, population members' placements are updated by dynamic simulations, which require extensive expertise for young bird growth. As young birds grow quickly, SFOA members update their positions and learn from experience to simulate their role in problem-solving and ensure the population's survival. To enhance the exploration capability of the global search algorithm, learning from experience is a set of process motions designed to generate meaningful shifts in SFOA members' locations. SFOA will use formula (9) to determine each member's position in order to improve the objective function.

$$Y_{new_{j,k}} = Y_{j,k}^y + L_d (lb + (ub - lb) \times rand), \quad t > 0.5 \quad (9)$$

Here, $Y_{new_{j,k}}$ indicates the position after population renewal, L_d is denoted as the size of the convolutional kernel, $Y_{j,k}^y$ indicates the location of the j superb fairywren in k dimension after y iterations and $rand$ is an arbitrary value between [0, 1].

Step 5: Breeding and Feeding Stage for Optimizing θ_1

In order to update population positions, SFOA replicated the breeding and nursing mechanisms of fantastic fairy-wrens. In order to avoid invasion by alien species, wrens will breed and employ special paternity tests during incubation when the danger threshold is low. Equation (10) shows the procedure for calculating hazard thresholds.

$$d = t1 * 20 + t2 * 20 \quad (10)$$

Here, $t1$ and $t1$ are arbitrary numbers with a normal distribution. As SFOA members mature, their activities become more diverse. A model of position change during egg breeding instruction was used to determine each SFOA member's new position using formula (11).

$$Y_{new_{j,k}} = Y_H + (Y_b Y_{j,k}^y) \times q, \quad t < 0.5 \quad (11)$$

Here, q define a factor, Y_H denotes breeding of the teaching eggs.

Step 6: Avoiding Natural Enemies Stage

When SFOA is in its predator avoidance phase, members update positions based on superb fairy-wren defense. Targeted members escape with slight position changes, while others hover, causing larger shifts. This dual strategy enhances the algorithm’s search range and local optimization, improving problem-solving efficiency, as described by mathematical formula (12).

$$Y_{new_{j,k}} = Y_b + Y_{jk} \times l \times k, \quad t < 0.5 \tag{12}$$

Here, l is the random step size of Levy flight, the control algorithm leaves the local optimal, k denotes the factor of adaptive flight balance.

Step 7: Termination

SFOA is used to optimize the weight parameter value of generators L_d and θ_1 from DBGNN. It then repeats step 3 until it achieves its halting criteria, $Y = Y + 1$. Then, by improving accuracy and cutting down on computation time, ADC-PBYBD-OL-DBGNN evaluates the detection of onion leaf disease.

4. Result and Discussion

The experimental findings of proposed ADC-PBYBD-OL-DBGNN method are deliberated in this part. The proposed technique is replicated utilizing MATLAB R2017a program and an Intel Core i52410M 2.3 GHz processor. Performance metrics like accuracy, recall, F1 score, precision, specificity, and computational time are used in the proposed approach. Obtained outcome of the proposed ADC-PBYBD-OL-DBGNN approach is analyzed with existing techniques including (OPLM-CD-Xpection) [11], (SAG-PDD-DNN) [12] and (OMI-PD-CNN) [13] respectively. Figure 4 displays output of ADC-PBYBD-OL-DBGNN system.









| <i>Dataset</i> | <i>Input Image</i> | <i>Preprocessed Image</i> | <i>Classification</i> |
|--|---|--|--------------------------|
| <i>Onion leaf dataset gathered from Chilwadigi village</i> |  |  | <i>Healthy</i> |
| |  |  | <i>Iris yellow virus</i> |
| |  |  | <i>Purple Blotch</i> |
| |  |  | <i>Leaf Blight</i> |

Figure 4: Output Result of Proposed ADC-PBYBD-OL-DBGNN method

4.1 Performance Measures

One often used indicator to assess a classification method's overall performance is accuracy. Accuracy is the proportion of correctly predicted events, such as true positives and true negatives, to all instances in a dataset. Using equation (13) the formula is derived.

$$Accuracy = \frac{TP + TN}{FN + TP + FP + TN} \quad (13)$$

Where, TP indicates True Positive; TN represents True Negative; FP represents False Positive and FN represents False Negative.

4.1.2 Precision

Precision refers to the degree of exactness and consistency in obtaining correct findings, whether in measurement detection. It displays the ability to consistently deliver results that match certain standards with minimum variability, ensuring dependability and detail in a variety of settings. Then the formula is derived in equation (14).

$$Precision = \frac{TP}{(TP + FP)} \quad (14)$$

4.1.3 Recall

Recall is defined as the ability to change in a system. It is frequently defined as the smallest change that accurately detect and higher recall indicates that a system classify smaller changes. Thus it is given in equation (15).

$$Recall = \frac{TP}{FN + TP} \quad (15)$$

4.1.4 F1 score

The F1score is a detection metric that balances recall and precision to assess a method's accuracy. It is computed by taking the harmonic mean of accuracy true positives among anticipated positives and recall. An F1-score of one denotes ideal performance, whereas a score of zero indicates poor performance. Thus it is given in equation (16).

$$F1Score = 2 \times \frac{Precision \times recall}{Precision + Recall} \quad (16)$$

4.2 Performance Analysis

Figures 5-7 and Table 3 and 4 portray the simulation results of ADC-PBYBD-OL-DBGNN method. Here, the proposed is STC-DGRGNN-HR-MRI analyzed with other OPLM-CD-Xpection, SAG-PDD-DNN and OMI-PD-CNN model correspondingly.

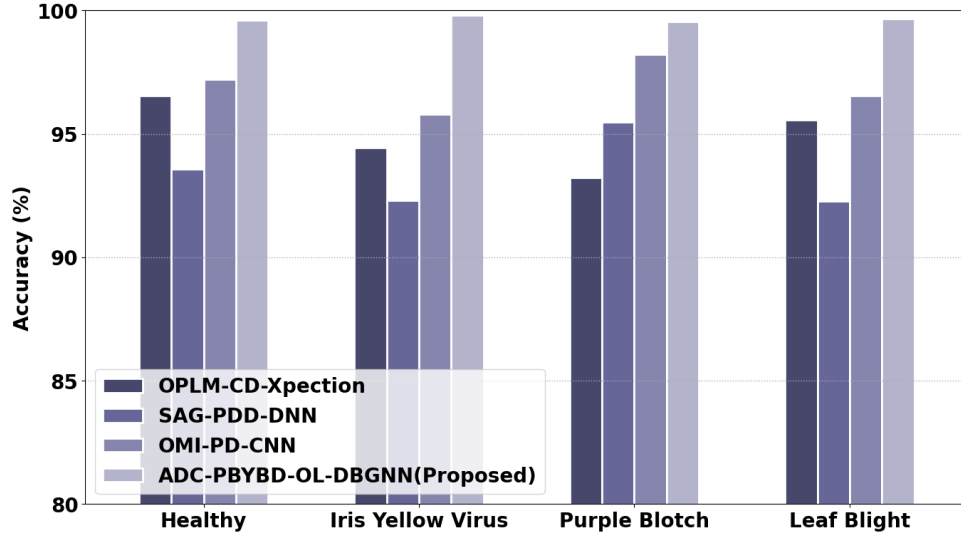


Figure 5: Performance Analysis of Accuracy

Figure 5 displays the proposed model Accuracy in achieving the desired results. The proposed ASP-ABSA-SDRGNN-MTF attains 26.68%, 25.75%, and 26.16% higher accuracy for Healthy, 27.31%, 28.47%, and 27.54% higher accuracy for Iris Yellow Virus, 30.73%, 28.35% and 29.62% higher accuracy for Purple Blotch, 27.34%, 23.97%, and 26.28% higher accuracy for Leaf Blight, comparing to the existing OPLM-CD-Xpection, SAG-PDD-DNN and OMI-PD-CNN model correspondingly.

Table 3: Performance Analysis of Precision

| Methods | Precision (%) | | | |
|--------------------------------------|---------------|-------------------|---------------|--------------|
| | Healthy | Iris Yellow Virus | Purple Blotch | Leaf Blight |
| OPLM-CD-Xpection | 96.44 | 94.32 | 92.65 | 93.46 |
| SAG-PDD-DNN | 93.53 | 91.5 | 96.56 | 91.25 |
| OMI-PD-CNN | 95.32 | 96.3 | 95.25 | 96.56 |
| ADC-PBYBD-OL-DBGNN (Proposed) | 98.22 | 98.63 | 98.34 | 98.25 |

Table 3 depicts the proposed model precision in achieving the desired results. The proposed ADC-PBYBD-OL-DBGNN attains 27.49%, 24.75%, and 25.85% higher precision for Healthy, 24.30%, 25.58% and 29.23% higher precision for Iris Yellow Virus, 25.23%, 30.46%, and 29.35% higher precision for Purple Blotch, 25.67%, 23.58%, and 31.45% higher precision for Leaf Blight, comparing to the existing OPLM-CD-Xpection, SAG-PDD-DNN and OMI-PD-CNN model correspondingly.

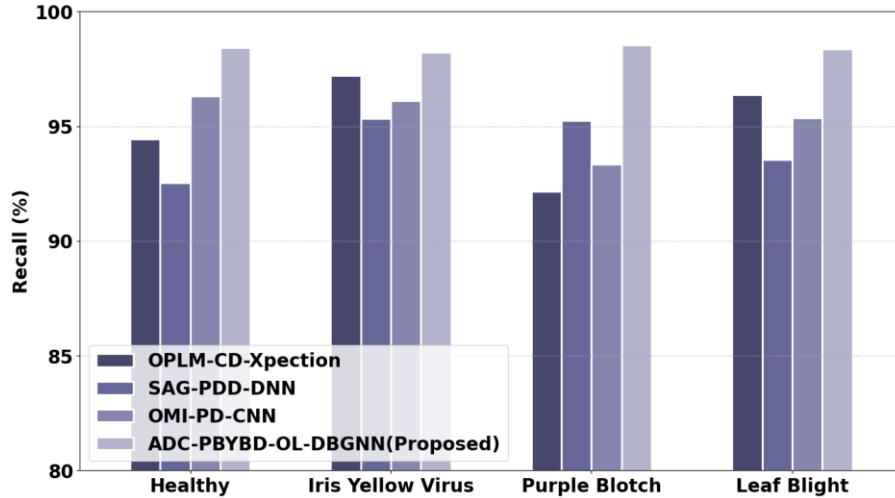


Figure 6: Performance Analysis of Recall

Figure 6 depicts the proposed model Accuracy in achieving the desired results. The proposed ADC-PBYBD-OL-DBGNN attains 29.08%, 30.70%, and 16.26% higher Recall for Healthy, 17.61%, 28.45%, and 23.54% higher Recall for Iris Yellow Virus, 25.73%, 28.05% and 24.62% higher Recall for Purple Blotch, 28.84%, 27.77%, and 25.08% higher Recall for Leaf Blight, comparing to the existing OPLM-CD-Xpection, SAG-PDD-DNN and OMI-PD-CNN model respectively.

Table 4: Performance Analysis of F1 score

| Methods | F1 score (%) | | | |
|---------------------------------------|--------------|-------------------|---------------|--------------|
| | Healthy | Iris Yellow Virus | Purple Blotch | Leaf Blight |
| OPLM-CD-Xpection | 97.34 | 94.32 | 96.55 | 92.26 |
| SAG-PDD-DNN | 93.23 | 96.54 | 93.46 | 95.35 |
| OMI-PD-CNN | 95.3 | 95.42 | 95.25 | 93.22 |
| ASP-ABSA-SDRGNN-MTF (Proposed) | 98.22 | 98.43 | 98.65 | 98.15 |

Table 4 depicts the proposed model F1 score in achieving the desired results. The proposed ADC-PBYBD-OL-DBGNN attains 25.49%, 23.53% and 24.27% higher F1 score for Healthy, 28.84%, 27.77%, and 25.08% higher F1 score for Iris Yellow Virus, 29.48%, 22.79%, and 25.82% higher F1 score for Purple Blotch, 26.28%, 24.49%, and 19.56% higher F1 score for Leaf Blight, comparing to the existing OPLM-CD-Xpection, SAG-PDD-DNN and OMI-PD-CNN model respectively.

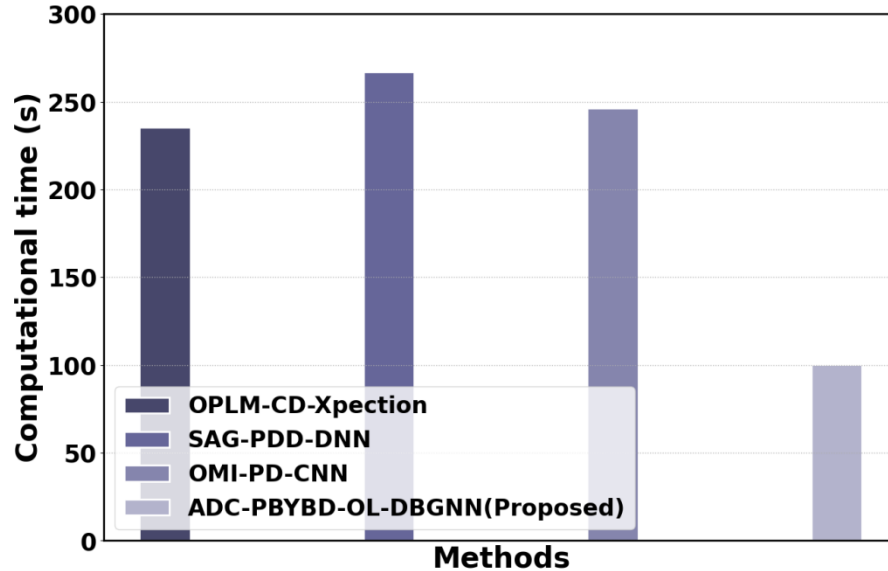


Figure 7: Performance Analysis of Computational Time

Figure 7 depicts the proposed model Computational Time in achieving the desired results. The proposed ADC-PBYBD-OL-DBGNN attains 29.48%, 22.79%, and 25.82% lower Computational time compared to the existing OPLM-CD-Xpection, SAG-PDD-DNN and OMI-PD-CNN model respectively.

5. Conclusion

In this section, ADC-PBYBD-OL-DBGNN is successfully executed. The study provides a dataset of onion plant leaf images that captures numerous diseases such as normal, Iris Yellow Virus, Purple Blotch, and Leaf Blight. This dataset gives critical insights into the issues of onion agriculture, serving as a dependable resource for researchers and agronomists seeking to evaluate disease patterns and develop focused methods for management. The performance of ADC-PBYBD-OL-DBGNN approach attains 26.68%, 25.75%, and 26.16% higher accuracy, 27.49%, 24.75%, and 25.85% higher precision when analyzed with existing methods like OPLM-CD-Xpection, SAG-PDD-DNN and OMI-PD-CNN respectively. Onion plants exhibit varying disease symptoms across stages, complicating detection. Environmental factors like humidity and temperature further impact disease appearance, making diagnosis more challenging. Future advancements in IoT and blockchain can enhance onion disease detection by enabling real-time monitoring of environmental factors and ensuring secure, tamper-proof disease diagnosis records.

Reference

1. Mahadevan, K., Punitha, A. and Suresh, J., 2024. Automatic recognition of rice plant leaf diseases detection using deep neural network with improved threshold neural network. *e-Prime-Advances in Electrical Engineering, Electronics and Energy*, 8, p.100534.
2. Hari, P., Singh, M.P. and Singh, A.K., 2024. An improved federated deep learning for plant leaf disease detection. *Multimedia Tools and Applications*, 83(35), pp.83471-83491.
3. Rashid, R., Aslam, W., Aziz, R. and Aldehim, G., 2024. An early and smart detection of corn plant leaf diseases using IoT and deep learning multi-models. *IEEE Access*, 12, pp.23149-23162.
4. Reis, H.C. and Turk, V., 2024. Potato leaf disease detection with a novel deep learning model based on depthwise separable convolution and transformer networks. *Engineering Applications of Artificial Intelligence*, 133, p.108307.
5. Rani, L., Sarangi, P.K. and Sahoo, A.K., 2024, February. Image-Feature Based Deep Learning Model for Plant Leaf Disease Detection. In *Macromolecular Symposia* (Vol. 413, No. 1, p. 2200216).
6. Khanna, M., Singh, L.K., Thawkar, S. and Goyal, M., 2024. PlaNet: a robust deep convolutional neural network model for plant leaves disease recognition. *Multimedia Tools and Applications*, 83(2), pp.4465-4517.
7. Dubey, R.K. and Choubey, D.K., 2024. An efficient adaptive feature selection with deep learning model-based paddy plant leaf disease classification. *Multimedia Tools and Applications*, 83(8), pp.22639-22661.
8. Sharma, V., Tripathi, A.K., Daga, P. and Mittal, H., 2024. CIGanNet: A novel method for maize leaf disease identification using CIGan and deep CNN. *Signal Processing: Image Communication*, 120, p.117074.
9. Kusuma, S. and Jothi, K.R., 2024. Early betel leaf disease detection using vision transformer and deep learning algorithms. *International Journal of Information Technology*, 16(1), pp.169-180.

10. Kaur, P., Mishra, A.M., Goyal, N., Gupta, S.K., Shankar, A. and Viriyasitavat, W., 2024. A novel hybrid CNN methodology for automated leaf disease detection and classification. *Expert Systems*, 41(7), p.e13543.
11. Aishwarya, M.P. and Reddy, A.P., 2024. Dataset of chilli and onion plant leaf images for classification and detection. *Data in Brief*, 54, p.110524.
12. Verma, S., Kumar, P. and Singh, J.P., 2024. MLNAS: Meta-learning based neural architecture search for automated generation of deep neural networks for plant disease detection tasks. *Network: Computation in Neural Systems*, pp.1-24.
13. Appiah, O., Hackman, K.O., Diakalia, S., Codjia, A.K.D., Bêbê, M., Ouedraogo, V., Diallo, B.A.A., Gandji, K., Abdoul-Karim, D., Ogunjobi, K.O. and Dabire, G., 2025. TOM2024: Datasets of tomato, onion, and maize images for developing pests and diseases AI-based classification models. *Data in Brief*, 59, p.111357.
14. Saraswathi, E. and Banu, J.F., 2024. Hybrid CGAN-based plant leaf disease classification using OTSU and surf feature extraction. *Neural Computing and Applications*, 36(23), pp.14395-14407.
15. Chai, A.Y.H., Lee, S.H., Tay, F.S., Goëau, H., Bonnet, P. and Joly, A., 2025. PlantAIM: A New Baseline Model Integrating Global Attention and Local Features for Enhanced Plant Disease Identification. *Smart Agricultural Technology*, p.100813.
16. <https://data.mendeley.com/datasets/tf9dtfz9m6/2>
17. Hovine, C. and Bertrand, A., 2025. Distributed adaptive spatial filtering with inexact local solvers. *IEEE Transactions on Signal Processing*.
18. Zhu, W., Zhou, X., Lan, S., Wang, W., Hou, Z., Ren, Y. and Pan, T., 2025. A dual branch graph neural network based spatial interpolation method for traffic data inference in unobserved locations. *Information Fusion*, 114, p.102703.
19. Jia, H., Zhou, X., Zhang, J. and Mirjalili, S., 2025. Superb Fairy-wren Optimization Algorithm: a novel metaheuristic algorithm for solving feature selection problems. *Cluster Computing*, 28(4), p.246.