

# Development of a Plant Disease Classification System using an Improved Counter Propagation Neural Network

B.O. Ola

Computer Engineering Dept.  
Osun State Polytechnic Iree,  
Osun State, Nigeria

J.P. Oguntoye

Computer Science and  
Engineering Dept.  
LAUTECH Ogbomosho,  
Oyo State Nigeria

O.O. Awodoye

Computer Science and  
Engineering Dept.  
LAUTECH Ogbomosho,  
Oyo State Nigeria

M.O. Oyewole

Centre for Information and  
Data Management.  
The Polytechnic Ibadan  
Oyo State Nigeria

## ABSTRACT

Plant diseases are a major threat to food security and can be precisely and accurately recognized through the images of plant leaves. The recent advances in computer vision made possible by the various computational method have paved the way for computer-assisted disease diagnosis. Thus, automated recognition of diseases on leaves plays a crucial role in the agriculture sector. Counter Propagation Neural Network (CPN) is highly desirable because it comprises the advantages of supervised and unsupervised training approaches. CPN in most image processing application guarantee high accuracy but consume more time for convergence. In this study, the development of a plant disease classification system using an improved Counter Propagation Neural Network (CPN) technique was carried out. Gravitational Search Algorithm (GSA) was applied to optimize the network of CPN for improved performance. The approach adopted in this study enhances CPN by making it free from the iterative adjustment of weights which increases the computational speed to a higher extent. The experimental results reveal that the proposed technique achieved improved performance in terms of recognition accuracy and prediction time.

## General Terms

Machine Learning, Image Processing, Agriculture.

## Keywords

Counter Propagation Neural Network, Gravitational Search Algorithm, Plant Diseases, Image Processing.

## 1. INTRODUCTION

In recent years, the rate of growth in agricultural production has been declining. This trend, coupled with emerging challenges such as climate change, population growth, rural to urban migration and demand for biofuels pose a serious challenge to global food security [16]. Food security is threatened by an alarming increase in the number of outbreaks of pests and plant diseases. These diseases jeopardize food security and have broad economic, social, and environmental impacts. Expertise in plant disease is necessary for an individual to be able to identify the diseased leaves [26].

Therefore, Fast and precise plant disease detection is very crucial to achieving an increased agricultural production in a sustainable manner. Generally, human specialists have been depended upon to detect defects in plants instigated by pests, diseases, malnutrition or severe climate. Nevertheless, the use of human expert is costly, tedious and time consuming and in some cases unfeasible. To overcome most of these issues, investigation into the utilization of image processing methods for plant disease detection has turn out to be a hot research

topic [21].

Specifically, many image processing and soft computing-based approaches have been proposed to detect plant leaf disease. Plants infected with diseases usually exhibit visible marks or lesions on either the leaves, stems, flowers and/or fruits. Generally, each disease or pest condition presents a unique visible pattern which can be used to uniquely diagnose the anomaly [21]. With the development of computer vision technique, researchers have proposed some plant disease recognition algorithms based on machine learning methods [15, 30, 2, 9; 29]. Nevertheless, the classification features in these approaches are selected based on human experience, which limits the generalizability of the models and the accuracies of these models do not still satisfy the recognition requirement.

Artificial Neural Networks (ANNs) are one of the widely used automated techniques. Though they yield high accuracy, most of the neural networks are computationally heavy due to their iterative nature. Counter Propagation Neural Network (CPN) is a hybrid neural network employing both supervised and unsupervised training methodologies [10]. A CPN consists of three layers: the input layer, the competition layer and the output layer. CPN in most image processing application guarantee high accuracy but consume more time for convergence. Since CPN have adaptable parameters, an optimization step is required to choose the most suitable network architecture. To make accurate predictions, CPNs require selecting an appropriate network size and training cycle (epochs) [4]. In this study, Gravitational Search Algorithm (GSA) an optimization algorithm based on the law of gravity and Newton's second law [17] was applied to optimize the network of CPN for improved performance in the classification of plant disease. GSA technique adopted in this study enhances CPN by making it free from an iterative adjustment of weights which increases the computational speed to a higher extent.

## 2. LITERATURE REVIEW

### 2.1 The Counter-Propagation Network

The counter-propagation network (CPN) started to be implemented in statistical analysis, function approximation and pattern classification, quite recently, since it has a moderately basic network structure and doesn't have an error benchmark for convergence due to combining Kohonen self-organizing map and Grossberg competitive learning network model [13, 14]. However, the CPN model often needs complex data storage memory and additional processing time than other neural network models. Therefore, for any CPN model that is utilized in practical engineering problems, it is

getting progressively critical and fundamental to implement competent learning algorithms and decrease the spatial dimensions of the measured data gotten from the plant dataset that is of huge and complex structures [7].

### The architecture of a CPN model

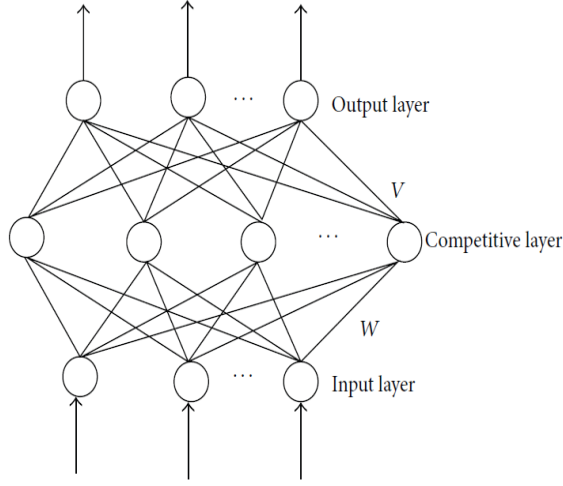


Fig 1: Architecture of a CPN model [11]

Generally, a CPN model comprises of three layers (as depicted in Fig 1), precisely, input layer, competitive layer, and output layer. The initial two layers is composed of self-organizing maps (SOMs). The SOM accomplishes the mapping of input data into two-dimensional plane, and as a rule, neurons are patterned in a rectangular or hexagonal grid. The last two layers constitute the Grossberg network which is generally known to be the competitive network model [4, 8].

From the input layer to the output layer, the winning neuron of competitive layer is gotten in agreement with the self-organization mapping (SOM) learning standard, and afterward the connecting weights sandwiched between the input layer and the competitive layer are adjusted utilizing the following learning rule [14]:

$$W_g(new) = W_g(old) + \alpha[A - W_g(old)] \quad (1)$$

Where  $W_g$  is the weight joining the winning neuron  $g$  in the competitive layer to individual neuron in the input layer;  $\alpha$  is the learning rate;  $A$  is the input vector. From the competitive layer to the output layer, the actual output value of each output neuron is attained in agreement with the learning rule of a general competitive network, and the connecting weights between them are set with respect of the supervised learning algorithm. Then the weight vectors are adapted by the following equation [14, 11]:

$$V_g(new) = V_g(old) + \beta b_g[C - V_g(old)] \quad (2)$$

Where  $V_g$  is the weight linking the winning neuron  $g$  in the competitive layer to neurons in the output layer;  $b_g$  is the output value of neuron  $g$ , with a value of 1;  $C$  is the desired output vector equivalent to the input vector  $A$ ;  $\beta$  is the learning rate. CPN has been broadly utilized in the fields of decision-making optimization, pattern recognition and robot intelligence for its flexibility and strength. Additionally, CPN is a combined network of both unsupervised and supervised learning; hence, it has the advantages of both teacher type and nonteacher type networks, like less iterations of data set, simpler network structure, and no error requirement for

convergence [13]. Nevertheless, the CPN model always needs considerable huge memory and computation time than other artificial neural network models, and it also stand the risk of failing in its learning procedure when a similar weight value is adjusted for test samples having comparable values [11, 14].

## 2.2 The Gravitational Search Algorithm

Gravitational Search Algorithm (GSA) is a population optimization algorithm based on the law of gravity and Newton's second law [24]. The algorithm searches for the optimal solution by moving the particle position of the population. That is, as the algorithm iterates, the particles move continuously in the search space by the gravitation between them [17]. In the same way as other nature-inspired algorithms, GSA needs refinements to improve its performance in resolving numerous kinds of problems. Notwithstanding, problem encoding can be sometimes challenging. Tuning the parameters of GSA play a major role in balancing the search time against the solution quality [28]. GSA is an optimization technique and offers appropriate balancing between exploitation and exploration functionalities. So, in this optimization algorithm, weightier masses agents are in charge of exploitation while the lighter masses agents are in charge of the exploration of the search space [1]. At the inception of the search process, lighter masses agents move with big step magnitude (exploration) and after this, once agent converge to the optimum solutions i.e. weightier masses agents interchange with proportional small step size (exploitation) [22]. The GSA algorithm is described by the following steps [24]:

**Step 1:** Initialize the speed and position of random agents and calculate the fitness of each agent.

The positions of the  $N$  number of agents are initialized randomly.

$$X_i = (x_i^1, \dots, x_i^d, \dots, x_i^n) \text{ for } i = 1, 2, \dots, N \quad (3)$$

$x_i^d$  represents the positions of the  $i^{th}$  agent in the  $d^{th}$  dimension, while  $n$  is the space dimension.

The fitness evolution is performed by evaluating the best ( $B(t)$ ) and worst ( $W(t)$ ) fitness for all agents at each iteration.

$$B(t) = \min \text{fit}_j(t) \quad j \in 1, \dots, N \quad (4)$$

$$W(t) = \max \text{fit}_j(t) \quad j \in 1, \dots, N \quad (5)$$

**Step 2:** Calculate the gravitational constant  $G(t)$  and the inertia mass of each particle:

$$G(t) = G_0 e^{(-\alpha t/T)} \quad (6)$$

$G_0$  and  $\alpha$  are initialized at the beginning and will be reduced with time to control the search accuracy.  $T$  is the total number of iterations.

$$M_{aj} = M_{pi} = M_{ii} = M_i \quad i = 1, 2, \dots, N \quad (7)$$

$$m_i(t) = \frac{\text{fit}_i - W(t)}{B(t) - W(t)} \quad (8)$$

$$M_i = \frac{m_i(t)}{\sum_{j=1}^N m_j(t)} \quad (9)$$

Where  $M_{ii}$  and  $M_{pi}$  are inertia and passive gravitational masses of  $i^{th}$  agent respectively and  $M_{aj}$  is an active gravitational mass of  $j^{th}$  agent.  $\text{fit}_i$  is the fitness value of  $i^{th}$  an agent.

**Step 3:** Calculate the resultant agent force, which can be given as:

$$F_{ij}^d(t) = G(t) \cdot \left( \frac{M_{pi}(t) \times M_{aj}(t)}{R_{ij}(t) + \epsilon} \right) \cdot (x_j^d(t) - x_i^d(t)) \quad (10)$$

where  $F_{ij}^d(t)$  is gravitation with the agent  $i$  and  $j$ , with

dimension,  $d$ , at the  $t$  generation;  $M_{pi}(t)$  is the passive gravitational mass related to an agent  $i$ ;  $\varepsilon$  is the small constant;  $x_i^d(t)$  and  $x_j^d(t)$  is the position of dimension,  $d$  of agent  $i$  and  $j$  at the  $t$  generation;  $R_{ij}(t) = \|x_i(t), x_j(t)\|_2$  is the Euclidean distance between agent  $i$  and  $j$ .

**Step 4:** Calculate accelerated speed. According to Newton's second law, the acceleration is obtained as follows:

$$a_i^d(t) = F_i^d(t)/M_{ii}(t) \quad (11)$$

$F_i^d(t)$  is the total force acting on  $i^{th}$  agent calculated as:

$$F_i^d(t) = \sum_{j \in Kbest, j \neq i} rand_j F_{ij}^d(t) \quad (12)$$

**Step 5:** Update speed and position:

The velocity update equation for agents is defined as

$$v_i^d(t+1) = rand_i \times v_i^d(t) + a_i^d(t) \quad (13)$$

$rand$  is random variable in interval  $[0,1]$ .  $v_i^d(t)$  and  $v_i^d(t+1)$  are the velocity of  $i^{th}$  agents during the iteration  $t$  and  $t+1$  respectively. The position update equation for agents is defined as:

$$x_i^d(t+1) = x_i^d(t) + v_i^d(t+1) \quad (14)$$

$x_i^d(t)$  and  $x_i^d(t+1)$  are the position of  $i^{th}$  agent during the iteration  $t$  and  $t+1$  respectively. The velocity of agents is updated during each iteration. Due to changes in the velocity every agent update its position.

**Step 6:** Check the termination condition. Terminate the optimization if the stopping criteria requirements are met, and, if not, compute the fitness and repeat the procedure from step 2 to 5 until the termination condition requirements are met.

## 2.3 Related Works

Detection and classification of plant diseases were attempted by Al-Hiary *et al.* [3] in 2011 using Artificial Neural Networks with 94% accuracy over 89.5% and 19% speedy computation time. Mainkar *et al.* [18] in 2015 employed K means clustering and ANN for plant disease detection and classification. The back-propagation algorithm was used in layered feed-forward ANNs by Patil *et al.* (2015).

Sladojevic *et al.* [27] in 2016, achieved an average accuracy of 96.3% in plant disease recognition and classification through CNN. Pujari *et al.* [23] in 2016, while comparing SVM and ANN in recognition and classification of plant diseases through image processing, found SVM to be more reliable than ANN, as the accuracies were 92.17% and 87.4%, respectively.

Kho *et al.* [12] in 2017, proposed an automated plant identification using artificial neural network and support vector machine. A total of 54 leaf image samples were used in this study. Evaluation results showed the ability of the proposed system to recognize leaf images with an accuracy of 83.3% with both SVM and ANN.

Rath and Meher [25] in 2019, proposed disease detection in infected plant leaf using a low complexity Radial Basis Function Neural Network (RBFNN) classifier. The detection of two most important diseases of rice i.e., brown spot and rice blast was done using RBFNN which achieved an accuracy, precision and recall of 95%, 97% and 95%, respectively.

## 3. METHODOLOGY

The scheme of the methodology used in this study is presented in Fig. 2.

### 3.1 Image Acquisition

The Flavia Leaf Dataset was used in this study. The Flavia Leaf Dataset contains 1907 scans of leaves belonging to 32 different species. The number of samples in each species varies from 50 to 77. This research employed some set of datasets out of the huge Flavia Leaf Dataset. Four species each with 200 leaf image samples (800 leaf datasets) was used out of each 480 datasets was grouped as train dataset while 320 datasets were assembled as test dataset. The image is shown in Fig. 3 (a).

### 3.2 Image Pre-processing

Pre-processing contains sequential operations. It included prescribing the image size, conversion of grey-scale images to thinning images and image normalization as shown in Fig. 3 (c). This phase was carried out to increase the quality of the image and remove the undesirable noise in image followed by clipping and smoothing of the image. The image enhancement is carried out to increase the contrast.

#### Conversion to Gray Scale

The RGB images are converted into grey images using colour conversion by the following formula in equation 15.

$$Gray(x) = \alpha R + \beta G + \gamma B \quad (15)$$

Where  $\alpha$ ,  $\beta$  and  $\gamma$  are the coefficient factor of Red, Green and Blue of the images. The output of the conversion is displayed in Fig 3 (b)

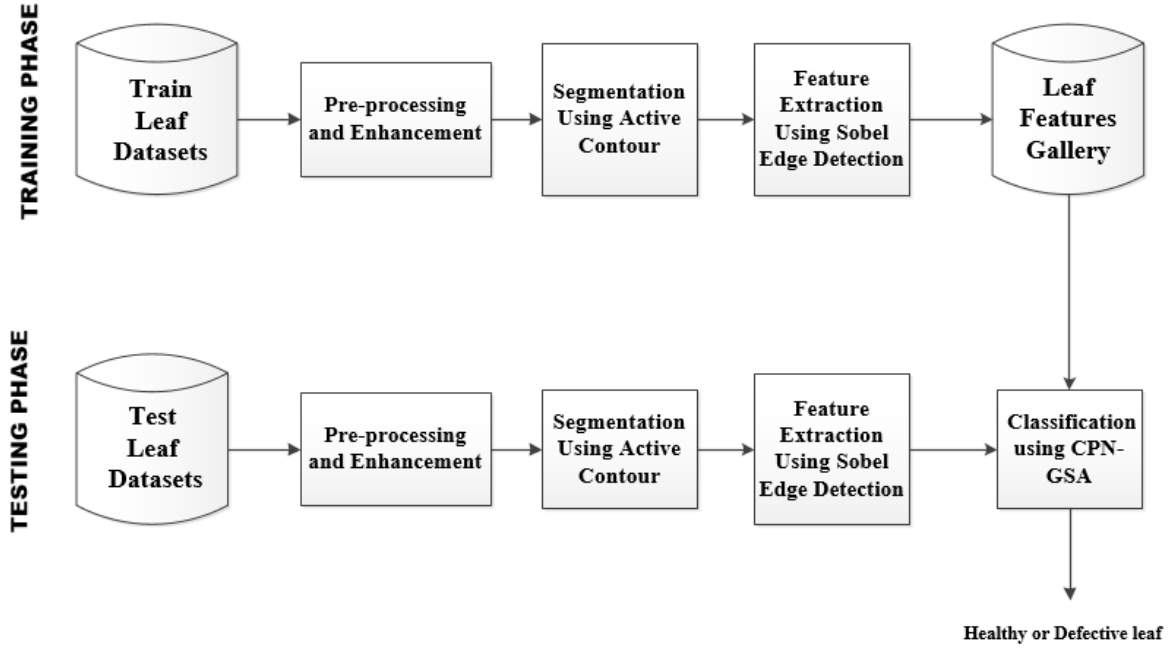


Fig. 2: The scheme of the methodology

### Image Normalization

Then Histogram Equalization (HE) is applied as image normalization in which the intensity of the image is distributed using cumulative distribution function. The cumulative distribution function is used to distribute the intensity values of the images. This function finds out the transformation that maps input image grayscale values to transformed image grayscale values.

### 3.3 Image Segmentation

Geometric active contour model was used in this study for segmentation. It is edge-based active contours which are closely related to the edge-based segmentation. Most edge-based active contour models consist of two parts: the regularity part, which determines the shape of contours, and the edge detection part, which attracts the contour towards the edges. Geometric active contour model was proposed by Caselles *et al.* [5] added the term, called stopping function, to the speed function shown in equation 16.

$$\frac{\partial \phi(x,y)}{\partial t} = |\nabla \phi(x,y)| (v + \varepsilon k(\phi(x,y))) \quad (16)$$

It was the first level set implemented active contour model for the image segmentation problem. Malladi *et al.* [19, 20] proposed a similar model given by

$$\frac{\partial \phi(x,y)}{\partial t} = g(I(x,y)) (k(\phi(x,y)) + v) |\nabla \phi(x,y)| \quad (17)$$

where  $g(\cdot): \Omega \rightarrow \mathbb{R}$  denotes the stopping function, i.e. a positive and decreasing function of the image gradient. A simple example of the stopping function is given by

$$g(I(x,y)) = \frac{1}{1 + |\nabla I(x,y)|^n} \quad (18)$$

where  $n$  is given as 1 in [5] and 2 in [20]. The contours move in the normal direction with a speed of

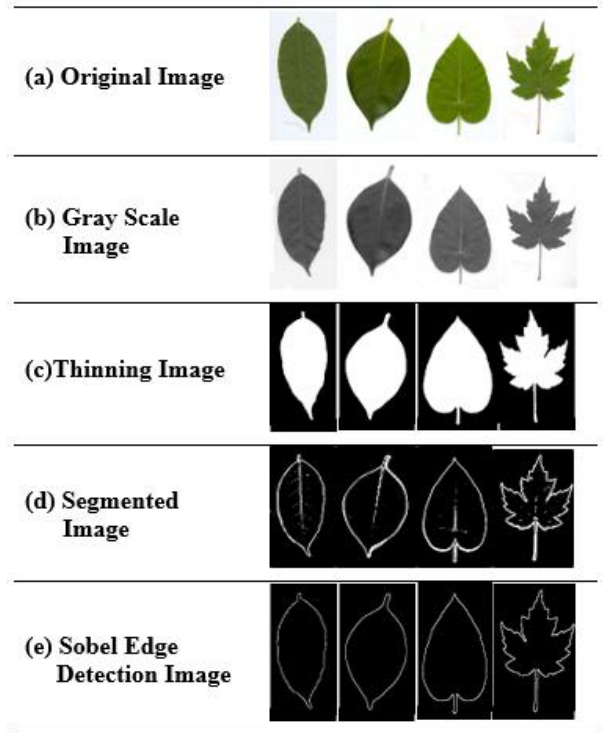


Fig 3

$g(I(x,y))(k(\phi(x,y)) + v)$ , and therefore stops on the edges, where  $g(\cdot)$  vanishes. The curvature term  $k(\cdot)$  maintains the regularity of the contours, while the constant term  $v$  accelerates and keeps the contour evolution by minimizing the enclosed area [6].

### 3.4 Feature Extraction using Sobel Edge Detection

Feature extraction is used to extract relevant features for recognition of plant leaves. The redundancy is removed from the image and the leaf images are represented by a set of

numerical features. These features are used to classify the data. The feature extraction employed in this study is Sobel Edge Detection.

Edge detection is more popular for identifying discontinuities in grey level than detecting isolated points and thin lines. The edge is the boundary between two regions with relatively distinct grey level properties. It is assumed here that the transition between the two regions can be properties. The transition between the two regions can be determined based on the grey level discontinuities. The Sobel operator accomplishes a 2-D spatial gradient estimation on an input image and thus highlights areas of high spatial frequency that match the edge of the images. The input image (grayscale), estimated gradient magnitude is as well recognized at individual point through the edge indicator. The operator comprises of a pair of 3x3 convolution kernels which is rotated by 90 degrees. The convolution masks of the Sobel detector are given in Fig. 4.

-1	0	+1
-2	0	+2
-1	0	+1

**Gx**

+1	+2	+1
0	0	0
-1	-2	-1

**Gy**

**Fig. 4: Sobel mask**

The procedure to achieve Sobel Edge Detection is as follow.

**Input:** A Sample Image.

**Output:** Detected Edges.

**Step 1:** Accept the input image.

**Step 2:** Apply mask Gx, Gy to the input image.

**Step 3:** Apply the Sobel edge detection algorithm and the gradient.

**Step 4:** Masks manipulation of Gx, Gy separately on the input image.

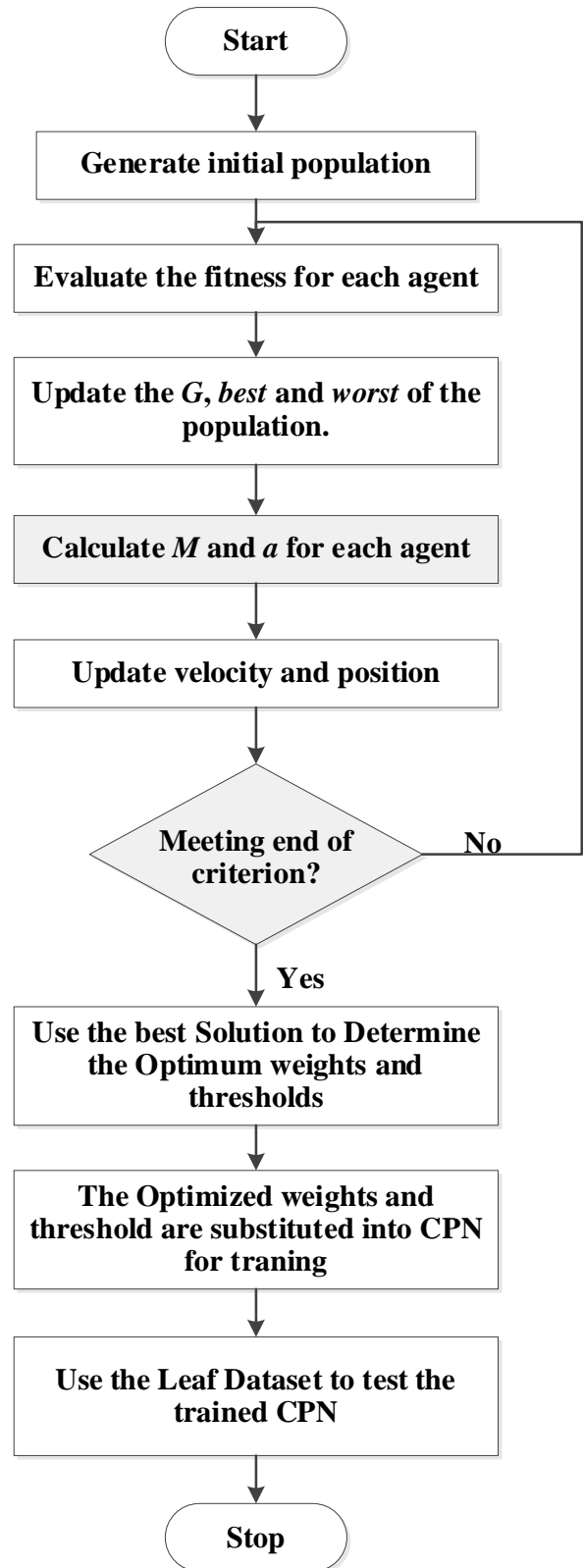
**Step 5:** Results combined to find the absolute magnitude of the gradient.

**Step 6:** The absolute magnitude is the output edges.

The output of feature extraction is as shown in fig. 3 (e).

### 3.5 Classification of Plant disease Using CPN-GSA Technique

The leaf image data matrix from the feature extraction stage was presented to the CPN-GSA technique for classification of a defect in leaf. GSA will be used to optimize the parameter of CPN for classification. The GSA technique will automatically choose the parameter for CPN for improved performance and minimum error. The steps for GSA used to optimized the parameter (weights, thresholds etc.) of CPN was stated in section 2.2 of this study while the principle of the CPN was stated in section 2.1 of this study. The initial position of the agents will be defined as stated in equation (3). Optimized weights and thresholds are used into CPN for disease prediction in plants through leaf images.



**Fig 5: The flowchart of CPN-GSA Prediction Model**

CPN prediction algorithm process is as follows:

**Step 1.** Load the data which is divided into training groups and testing groups, and initial processing, respectively.

**Step 2.** Initialize the population and iteration count of the GSA optimization algorithm, the positions of the agent in the

dimension and the space dimension.

**Step 3.** The optimal value is calculated by the GSA optimization algorithm.

**Step 4.** The optimized weights and thresholds are substituted into the CPN for training.

**Step 5.** The performance of the trained CPN is tested by using testing groups, and the error is calculated. Figure 3.3 depicts how the process GSA was used to optimize the parameter of CPN.

### 3.6 Evaluation Measures

The overall performance of the techniques under study was evaluated based on accuracy, False Positive Rate (FPR), Sensitivity and computation time. Confusion matrix was used to determine the values of the performance metrics. It contains “True Positive (TP), False Positive (FP), False Negative (FN) and True Negative (TN).” The relations are given as:

$$\text{False Positive Rate (FPR)} = \frac{FP}{TN+FP} \quad (19)$$

$$\text{Sensitivity} = \frac{TP}{TP+FN} \quad (20)$$

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN} \quad (21)$$

### 3.7 Implementation

MATLAB R2018 on Windows 10 64-bit operating system, Intel®Core™ i5-2540M CPU@2.60GHz Central Processing Unit, 6GB Random Access Memory and 500GB hard disk drive was used to implement the techniques. A total number of 32 leaf images were used to test the proposed technique.

## 4. RESULT AND DISCUSSION

The result of the classification technique considered in this study is presented in this section. **Table 1** presents the contingency table for CPN-GSA and CPN. The leaf dataset for testing the techniques comprises of 320 images out of which 160 were healthy leaves without defects and 160 were unhealthy leaves without defects.

**Table 2: Contingency table for classification using CPN-GSA and CPN Techniques**

Techniques		CPN-GSA		CPN	
		Predicted Class		Predicted Class	
		Non-defect	defect	Non-Defect	Defect
Actual Class	Non-Defect (93)	156 (TP)	4 (FN)	153 (TP)	7 (FN)
	Defect (187)	1 (FP)	159 (TN)	4 (FP)	156 (TN)

The result obtainable in **Table 1** revealed that the developed CPN-GSA technique correctly classified 156 of the healthy leaf datasets as healthy without defects while 4 of the datasets were misclassified as unhealthy with defects. Also, 159 of the unhealthy leaf datasets were correctly classified as unhealthy while 1 was wrongly classified as healthy without defects. Similarly, the CPN technique, correctly classified 153 of the healthy leaf datasets as healthy without defects while 7 of the datasets were misclassified as unhealthy with defects. Also, 156 of the unhealthy leaf datasets were correctly classified as unhealthy while 4 was wrongly classified as healthy without defects. This shows that the CPN-GSA classification

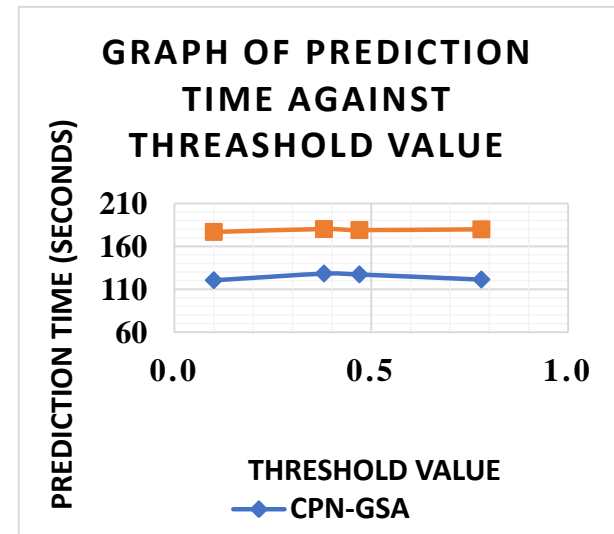
algorithm resulted in a notable reduction of the misclassification error.

**Table 2** depicts the performance of CPN-GSA and CPN technique for validation measures; false-positive rate, sensitivity, accuracy and prediction time at a threshold value of 0.78.

**Table 2: Results obtained by the CPN-GSA and CPN technique**

Technique	FPR (%)	Sensitivity (%)	Accuracy (%)	Prediction time (Sec)
CPN-GSA	0.63	97.50	98.44	121.17
CPN	2.50	95.63	96.56	179.63

The result obtainable in **Table 2** reveals that the CPN-GSA technique achieved FPR, sensitivity and accuracy of 0.63%, 97.50%, 98.44% at 121.17 seconds respectively. Similarly, the CPN technique has FPR, sensitivity and accuracy of 2.50%, 95.63%, 96.56% at 179.63 seconds respectively. The result shows that CPN-GSA technique has increased accuracy and sensitivity of 1.88% and 1.87% respectively and a decreased FPR and prediction time of 1.87% and 58.46 seconds respectively. The application of CPN with GSA gave an improved performance compared to the standard CPN in the classification of diseases in leaves. Hence, the CPN-GSA techniques outperformed the standard CPN in terms of FPR, sensitivity, accuracy and prediction time. The application of GSA to optimize the parameter of CPN for improved performance is justified. **Fig. 6** depicts the graph of prediction time against the threshold values.



**Fig 6: The graph of prediction time against the threshold values**

The results from the graph show that the application of GSA along with CPN techniques reduces the time complexity of the standard CPN. Hence, the CPN-GSA technique trains and classifies faster than the CPN technique.

From **Table 3**, Inferential statistical analysis using paired sampled t-test done to analyze the result obtained for both CPN-GSA and CPN technique for accuracy and prediction time reveal that the test of significance of the aforementioned metrics evaluated at 95% confidence level shows that there was a significant difference between CPN-GSA and CPN technique at  $P < 0.05$ . The t-test result validates the fact that

the CPN-GSA technique outperformed the CPN technique in terms of accuracy and prediction time in the classification of disease in leaves. Hence the CPN-GSA technique has significant improvement in accuracy and prediction time over the standard CPN classification.

**Table 4 Statistical comparison between CPN-GSA and CPN on paired sampled t-test at the 0.05 significance level**

Techniques	Measures	Mean difference	t value	p-value
CPN-GSA vs CPN	Accuracy	1.88	751.0	.000
	Prediction Time	-54.55	-32.168	.000

The improved performance achieved in this study is due to the modification of the original CPN classification algorithm using GSA which reduces the misclassification error in the region of uncertain decision. Also, the application of the GSA with the CPN model enhances the performance of CPN which sometimes fail in its learning process when the same weight value is adjusted for samples with similar values [11]. Also, it helps the CPN to limits the required storage memory and computing time to train and classify data. Ballabio *et al.*, [4] stated that the Counter-Propagation Artificial Neural Networks require an optimization step to choose the most suitable network architecture to make accurate predictions. The result achieved in this study corroborates the findings of Ballabio *et al.*, [4] who achieved improved performance by optimization of the parameter of CPN. The findings in this study, suggest a solution to the limitations in some existing works [25, 12, 27, 18] which achieved promising results but with high computational complexity.

Hence, this study shows that the integration of GSA and CPN has great potential in the field of plant disease detection. However, it is noted that the proposed model has been proved feasible and efficient only in the case of classification through leaves. The same can be applied for the classification of diseases through fruits, stems and flowers. Its actual performance in experimental work in the suggested areas needs to be studied in the coming investigations.

## 5. CONCLUSION

It is a fact that severe diseases in plants lead to the annual losses of the agricultural yield. Consequently, detecting the diseases at an early stage in plants is very crucial for the prevention of such drastic losses and also has a pivotal role in their management in the future. The proposed CPN-GSA technique was found to be more accurate, sensitive with less false positives rates and less computationally expensive. The approach used in this study can help both pathologists and farmers to detect the disease at an early stage and also correctly. It is evident that CPN-GSA technique is well matched to the other conventional classification method based on its performance. Hence, it will achieve a more accurate and computationally efficient Computer-Aided Detection and Diagnosis system in early detection of plant disease. Future work can be carried out to investigating the proposed approached on different types of Plants too to identify the disease in that particular plant.

## 6. REFERENCES

- [1] Aditi G., Nirmala S., & Harish S. (2017). Exploitative Gravitational Search Algorithm. Proceedings of Sixth International Conference on Soft Computing for Problem Solving, Advances in Intelligent Systems and Computing

546, DOI 10.1007/978-981-10-3322-315.

- [2] Akbarzadeh, S., Paap, A., Ahderom, S., Apopei, B., & Alameh, K. (2018). Plant discrimination by Support Vector Machine classifier based on spectral reflectance. *Computers and electronics in agriculture*, 148, 250-258.
- [3] Al-Hiary, H., Bani-Ahmad, S., Reyat, M., Braik, M., & Alrahmaneh, Z. (2011). Fast and accurate detection and classification of plant diseases. *International Journal of Computer Applications*, 17(1), 31-38.
- [4] Ballabio, D., Vasighi, M., Consonni, V., & Kompany-Zareh, M. (2011). Genetic algorithms for architecture optimisation of counter-propagation artificial neural networks. *Chemometrics and intelligent laboratory systems*, 105(1), 56-64.
- [5] Caselles, V., Catté, F., Coll, T., & Dibos, F. (1993). A geometric model for active contours in image processing. *Numerische mathematik*, 66(1), 1-31.
- [6] Cohen, F. S., & Fan, Z. (1992). Maximum likelihood unsupervised textured image segmentation. *CVGIP: Graphical models and image processing*, 54(3), 239-251.
- [7] Du, K. L. (2010). Clustering: A neural network approach. *Neural networks*, 23(1), 89-107.
- [8] Guang, C., Li, Q., & Yang, L. (2008). Research on the application of CPN neural network to fault diagnosis of analog circuits. *Ship Electronic Engineering*, 28(5), 5-7.
- [9] Hamuda, E., Mc Ginley, B., Glavin, M., & Jones, E. (2017). Automatic crop detection under field conditions using the HSV colour space and morphological operations. *Computers and electronics in agriculture*, 133, 97-107.
- [10] Hemanth, D. J., Vijila, C. K. S., & Anitha, J. (2010). Performance improved PSO based modified counter propagation neural network for abnormal MR brain image classification. *Int. J. Advance. Soft Comput. Appl*, 2(1), 65-84.
- [11] Jiang, S. F., Fu, C., Zhang, C. M., & Wu, Z. Q. (2013). A revised counter-propagation network model integrating rough set for structural damage detection. *International Journal of Distributed Sensor Networks*, 9(11):1-9.
- [12] Kho, S. J., Manickam, S., Malek, S., Mosleh, M., & Dhillon, S. K. (2017). Automated plant identification using artificial neural network and support vector machine. *Frontiers in Life Science*, 10(1), 98-107.
- [13] Kuzmanovski, I., & Novič, M. (2008). Counter-propagation neural networks in Matlab. *Chemometrics and Intelligent Laboratory Systems*, 90(1), 84-91.
- [14] Kuzmanovski, I., Novič, M., & Trpkovska, M. (2009). Automatic adjustment of the relative importance of different input variables for optimization of counter-propagation artificial neural networks. *Analytica chimica acta*, 642(1-2), 142-147.
- [15] Liu, B., Ding, Z., Tian, L., He, D., Li, S., & Wang, H. (2020). Grape Leaf Disease Identification Using Improved Deep Convolutional Neural Networks. *Frontiers in Plant Science*, 11, 1082.
- [16] Liu, P. (2015). The future of food and agriculture: Trends and challenges. *Food and Agriculture Organization of*



*the United Nations.*

- [17] Lu, P., Ye, L., Sun, B., Zhang, C., Zhao, Y., & Teng, J. (2018). A new hybrid prediction method of ultra-short-term wind power forecasting based on EEMD-PE and LSSVM optimized by the GSA. *Energies*, 11(4), 697.
- [18] Mainkar, P. M., Ghorpade, S., & Adawadkar, M. (2015). Plant leaf disease detection and classification using image processing techniques. *International Journal of Innovative and Emerging Research in Engineering*, 2(4), 139-144.
- [19] Malladi, R., & Sethian, J. A. (1995). Image processing via level set curvature flow. *proceedings of the National Academy of Sciences*, 92(15), 7046-7050.
- [20] Malladi, R., Sethian, J. A., & Vemuri, B. C. (1995). Shape modelling with front propagation: A level set approach. *IEEE transactions on pattern analysis and machine intelligence*, 17(2), 158-175.
- [21] Ngugi, L. C., Abelowahab, M., & Abo-Zahhad, M. (2020). Recent Advances in Image Processing Techniques for Automated Leaf Pest and Disease Recognition-A Review. *Information Processing in Agriculture*.
- [22] Ola B. O., Awodoye O. O. and Oguntoye J. P. (2019). A comparative study of particle swarm optimization and gravitational search algorithm in a poultry house temperature control system. *World Journal of Engineering Research and Technology (WJERT)*. 5(6): 272-289.
- [23] Pujari, D., Yakkundimath, R., & Byadgi, A. S. (2016). SVM and ANN-based classification of plant diseases using feature reduction technique. *IJIMAI*, 3(7), 6-14.
- [24] Rashedi, E., Nezamabadi-Pour, H., Saryazdi, S. (2009). GSA: A Gravitational Search Algorithm.” *Information sciences*, 179(13): 2232-2248.
- [25] Rath, A. K., & Meher, J. K. (2019). Disease detection in infected plant leaf by computational method. *Archives of Phytopathology and Plant Protection*, 52(19-20), 1348-1358.
- [26] Singh, D., Jain, N., Jain, P., Kayal, P., Kumawat, S., & Batra, N. (2020). PlantDoc: a dataset for visual plant disease detection. In *Proceedings of the 7th ACM IKDD CoDS and 25th COMAD* (pp. 249-253).
- [27] Sladojevic, S., Arsenovic, M., Anderla, A., Culibrk, D., & Stefanovic, D. (2016). Deep neural networks-based recognition of plant diseases by leaf image classification. *Computational intelligence and neuroscience*, 2016.
- [28] Taisir E. & Al Qasim R. (2013). On the Performance of the Gravitational Search Algorithm. *International Journal of Advanced Computer Science and Applications (IJACSA)*. Vol. 4, No. 8, pp. 74-78.
- [29] Waghmare, H., Kokare, R., & Dandawate, Y. (2016). Detection and classification of diseases of grape plant using opposite colour local binary pattern feature and machine learning for an automated decision support system. In *2016 3rd international conference on signal processing and integrated networks (SPIN)* (pp. 513-518). IEEE.
- [30] Wang, X., Zhu, C., Fu, Z., Zhang, L., and Li, X. (2019). Research on Cucumber Powdery Mildew Recognition Based on Visual Spectra. *Spectroscopy And Spectral Analysis*, 39(6), 1864-1869.