Tuijin Jishu /Journal of Propulsion Technology
ISSN:1001-4055
Vol. 44 No.04 (2023)

Deep Neural Network Classifier for Detecting Cotton Plant Diseases Using Antlion Optimization

T. Kalaiselvi[1], V. Narmatha[2]

[1]*Research Scholar, Department of Computer and Information Science, Annamalai University, Annamalainagar – 608 002, Tamil Nadu, India*

[2]*Assistant Professor, Department of Computer and Information Science, Annamalai University, Annamalainagar – 608 002, Tamil Nadu, India*

Abstract

Image processing techniques are used to autonomously identify agricultural plant diseases can significantly reduce reliance on farmers for safeguarding crop yields. The classification of cotton leaf diseases presents a formidable challenge. This study introduces a novel approach for classifying Cotton Leaf Diseases, employing an Antlion Optimization (ALO)-enhanced Deep Neural Network (DNN) classifier. The dataset comprises 10,000 images, a combination of directly captured farm field images and downloaded samples, encompassing normal leaves, bacterial blight, Anthracnose, Cercospora leaf spot, and Alternaria diseases. Preprocessing incorporates Wiener filtering to eliminate image noise, while Fuzzy Rough C-Means (FRCM) clustering is employed for diseased and normal portion segmentation. The ALO-augmented DNN achieves an impressive 93.37% accuracy in classifying cotton leaf diseases.

Keywords: *Classification, Antlion, Optimization, Cotton leaf, and Deep Neural Network*

1. Introduction

Image processing is a method to convert an image into digital form and perform some operations on it, in order to get an enhanced image or to extract some useful information from it. Image pre-processing techniques are used to improve the quality of an image before processing into an application [1]. This uses a small neighborhood of a pixel in an input image to get a new brightness value in the output image. These pre-processing techniques are also called filtration and resolution enhancement [2]. Most of the imaging techniques are degraded by noise. In order to preserve the edges and contour information of the Agricultural plant images, the efficient de-noising and an improved enhancement technique is required. Gaussian, Median and Wiener filters are used for image de-noising [3]. The agriculture sector can be considered as the backbone for any developing economy. To obtain the maximum yield from the crops, it is required that farmers should be provided with the best technologies and methodologies [4].

India stands second in world cotton production. In-spite of its huge production, the diseases affects the cotton plant. Diseases could be bacterial, viral or parasitic in nature. Common cotton diseases are bacterial blight, powdery mildew, leaf curl, boll rot, fusarium wilt, cercospora, anthracnose, alternariaetc[5]. By applying image processing and deep learning techniques will prove to be useful in detecting the above stated diseases. According to Piyush Singh, coconut trees were affected due to pest infection stem bleeding and leaf blight diseases[6]. A total of 1564 images were captured from the farms of kayar village, Tamilnadu. Using FRCM for segmentation and convolutional neural network as a classifier, he was able to achieve a validation accuracy of

96.94%. The paper describes the various architectures of CNN and compares the performance of each of them in accurate and detail [7].

In this work, section2 discusses the implementation of various filters such as Wiener, Median and Gaussian to remove the noise present in the cotton images. Section3 elaborates the FRCM segmentation technique to segment the cotton leaf diseased portion from the cotton leaf. In section4 Antlion based Deep Neural Network classifier is designed and implemented. Section5 deals with results and discussion of the proposed work. Section5 discusses the conclusion of the work.

2. Pre-Processing

The purpose of pre-processing is to enhance the quality of the image, remove noise, and extract relevant information to make subsequent processing steps more effective. Images captured from various sources, such as cameras or sensors, often contain unwanted noise, which can degrade the quality of the image. Pre-processing techniques like filtering can help reduce noise, making the image cleaner and more suitable for analysis [8].

In this work, Wiener filter, Median filter and Gaussian filters are used for noise removal in cotton images. Based on Mean Square Error (MSE) and Peak Signal to Noise Ratio (PSNR), best filter will be selected for further analysis. MSE is a widely used image quality metric, and it quantifies the average squared difference between pixel values in the original and filtered images. However, like PSNR, it's important to remember that MSE may not always capture the perceptual quality of an image accurately.

A low Mean Square Error (MSE) value indicates that the filtered image is very close to the original image, and there is minimal distortion or noise [10].A high MSE value indicates that the filtered image is significantly different from the original image, and there is a higher level of distortion or noise. For image quality assessment using the Peak Signal-to-Noise Ratio (PSNR), a higher PSNR value indicates better image quality. In other words, a higher PSNR means that the reconstructed or filtered image is more similar to the original image, and there is less distortion or noise in the processed image.

Wiener filters are a class of optimum linear filters which involve linear estimation of a desired signal sequence from another related sequence. It is not an adaptive filter. The Wiener filter's main purpose is to reduce the amount of noise present in an image by comparison with an estimation of the desired noiseless image. The Wiener filter may also be used for smoothing. This filter is the mean squares error-optimal stationary linear filter for images degraded by additive noise and blurring. It is usually applied in the frequency domain (by taking the Fourier transform), due to linear motion or unfocussed optics. Wiener filter is the most important technique for removal of blur in images.

$\hat{x}(m)$ - Least Mean Square Error (LMSE) to estimate a desired or expected signal $\hat{x}(m)$.The relation between input signal and output signal is expressed as

$$\hat{x}(m) = \sum_{k=0}^{P-1} [w_k * y(m-k)] \tag{1}$$

$$= w^T * y \tag{2}$$

where,

m - Discrete-time index

$y^T$is consider as filter input, and it is equal to [y(m), y(m–1), ..., y(m–P–1)]

w^T= [w0, w1, ..., wP–1] - Wiener filter coefficient vector.

The performance measures of cotton diseased images by implementing wiener, median and Gaussian filters are presented in Table1.

Table 1 performance measures of cotton diseased images by implementing filters.

Image	Wiener Filter		Median Filter		Gaussian filter	
	PSNR	**MSE**	**PSNR**	**MSE**	**PSNR**	**MSE**
1	30.21 dB	165.43	26.12 dB	519.92	22.58 dB	935.67
2	42.11 dB	125.87	34.46 dB	228.78	27.79 dB	464.73

Wiener filter, Median filter and Gaussian filter are tested for 100 cotton diseased images. Based on the PSNR and MSE, wiener filter produces better result when compare with those of Median filter and Gaussian filter. Therefore wiener filter is considered for pre-processing. The input images and pre processed images are shown from Fig.1 to Fig.4.


Fig.1 Input image1 Fig.2.Pre-processed image1 using wiener filter


Fig.3 Input image 2 Fig.4.Pre-processed image2 using wiener filter

3. Fuzzy Rough C-Means Algorithm for Cotton Disease Image Segmentation

The FRCM (Fuzzy Relational C-Means) algorithm commences by initializing a set of cluster centers, which serve as the prototypes for the clusters [8]. In the context of fuzzy clustering, each pixel within the image is allowed to belong to multiple clusters with different levels of membership. A membership function assigns a membership value to each pixel for each cluster, signifying the extent of its association with that cluster. FRCM's primary objective is to minimize a composite function that consists of two components. The first component quantifies the similarity between each pixel and the cluster centers, and the second component is determined by the membership values, which are computed based on the distances between pixels and the cluster centers.

Rough set theory serves as a valuable tool for managing the uncertainty and granularity inherent in data [11]. It proves particularly useful in addressing ambiguities or areas of noise within an image. Once the membership values are computed, the algorithm proceeds to revise the cluster centres by taking a weighted average of pixel values, with the weights determined by the membership values. This iterative process of updating both the cluster centers and membership values continues until a predefined termination criterion is satisfied. This criterion can take the form of a set number of iterations or a convergence threshold.

Upon algorithm convergence, each pixel is assigned to the cluster with the highest membership value. This assignment yields an image segmentation, grouping pixels with similar characteristics into the same cluster. FRCM harnesses the strengths of fuzzy clustering, which adeptly manages the inherent uncertainty in image data, and rough set theory [12], which is effective at addressing data granularity and noise. This hybrid approach can produce robust image segmentation, especially in scenarios where traditional clustering methods may falter due to noise or ambiguity.

FRCM introduces the notion of fuzzy membership within fuzzy sets and incorporates the concepts of lower and upper approximations from rough sets into the c-means algorithm. The utilization of fuzzy sets' membership facilitates the efficient handling of overlapping partitions, while the integration of rough sets deals with issues of uncertainty, vagueness, and incompleteness in class definitions. The FRCM algorithm partitions a set $X=(x_1 \ldots . x_i \ldots . x_n)$of n objects into c clusters by minimizing the following objective function. The segmented images by implementing FRCM are shown in Figs 6,8 and 10.

Fig.5PreprocessedAlternaria infected image

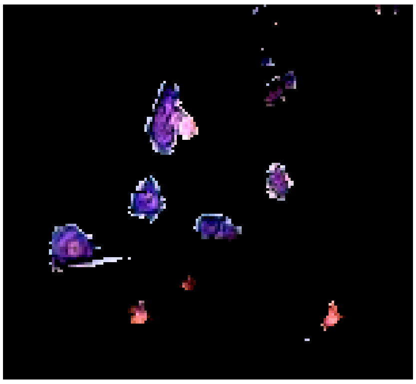

Fig.6 Segmented image using FRCMclustering

Fig.7 Preprocessed Cercospora Leaf Spot Leaf

Fig.8 Segmented image using FRCM clustering

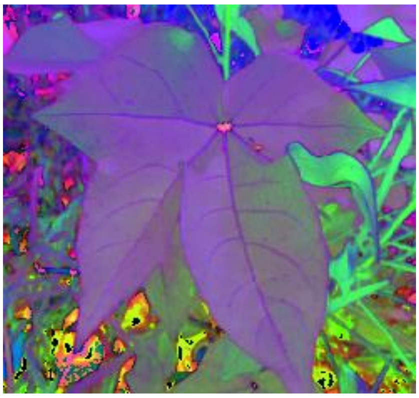

Fig.9 Preprocessed healthy Leaf **Fig.10 Segmented image using FRCM clustering**

4. Antlion Optimization

The Antlion Optimization Algorithm (ALO) is a nature-inspired optimization algorithm that is based on the hunting behavior of antlions, a type of insect commonly found in sandy environments [13]. The ALO algorithm is used to solve optimization problems by mimicking the way antlions create traps to capture their prey, primarily ants. Antlions are insects that inhabit sandy areas. They dig conical pits in loose sand and wait at the bottom to capture ants and other small insects that fall into the pit. The antlion's strategy is to optimize the location and shape of the pit in such a way that it maximizes the chances of catching prey.The ALO algorithm is based on the idea of optimizing solutions through a two-phase process: the random walk phase and the trap-building phase. These phases mimic the antlion's behavior in nature.

4.1 Random Walk Phase

In this phase, potential solutions (represented as candidate positions) are randomly generated. The random walk imitates the wandering movement of an ant searching for food. The quality of these random solutions is evaluated using an objective function, which measures their fitness in the context of the optimization problem.

4.2 Trap-Building Phase

The trap-building phase simulates the process by which antlions construct their pits. Antlions adjust their traps by removing or adding sand grains to the pit. In the ALO algorithm, candidate solutions are adjusted based on the fitness values obtained in the random walk phase. This adjustment helps to concentrate solutions in promising areas of the search space.

4.3 Iterative Process

ALO iteratively performs the random walk and trap-building phases. As the algorithm progresses, it refines the solutions and converges toward an optimal or near-optimal solution.

5. Antlion Optimization based Deep Neural Network Classifier

In the realm of artificial intelligence and computer vision, deep neural networks have revolutionized image classification. These networks, often referred to as Convolutional Neural Networks (CNNs) when applied to images, have become the cornerstone of modern computer vision tasks [14]. Deep neural networks are well-suited for image classification due to their ability to automatically learn hierarchical features and patterns from raw pixel data. They consist of input layer, multiple hidden layers and output layer. Training a deep neural network for image classification involves two critical steps: data collection and model training. A diverse and representative dataset of labeled images is required for the network to learn from. During training, the model fine-tunes its internal parameters through backpropagation, adjusting them to minimize the difference between its predictions and the ground truth labels [15].

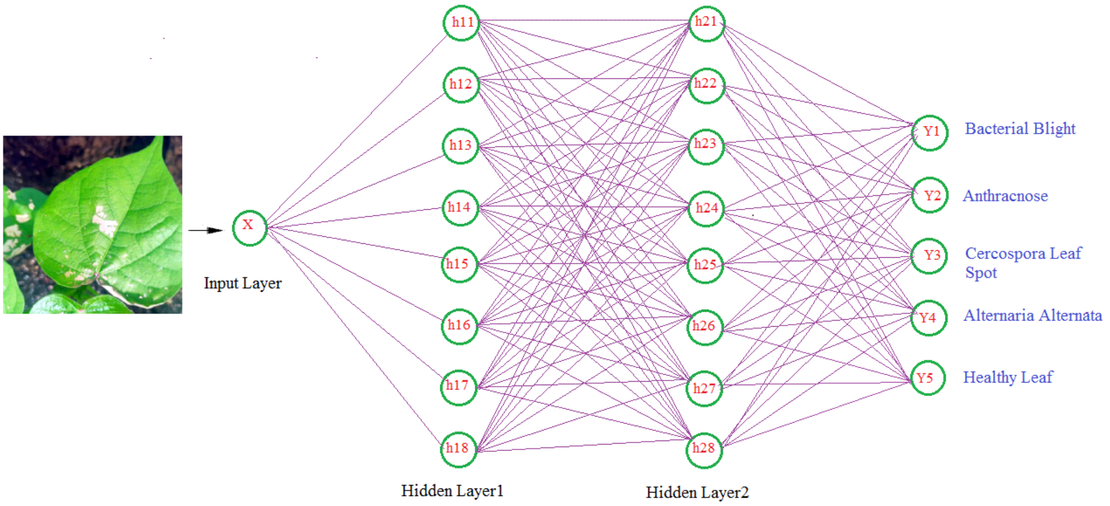

Fig.11. Architecture of DNN with two hidden layers.

The DNN architecture, as illustrated in Figure 11, is designed with two hidden layers to effectively capture the mapping relationship between input and output data, with careful consideration of preference weight fitness. During the training phase, the ALO-DNN dynamically adjusts the number of hidden neurons in these layers through iterative updates. As the training iterations progress, the neural network continuously adapts its decision boundary to better fit the labeled training data. To improve both the training speed of the ALO-DNN and its classification accuracy, two hidden layers are incorporated. The total number of nodes in these hidden layers is determined using a specific equation.

$$N = \sqrt{a + b} + c \qquad (4)$$

For enabling the non-linear fitness ability an activation function is added in the hidden layer of DNN. We have used the softmax as an activation function and it is given as,

$$S(y)_i = \frac{\exp(y_i)}{\sum_{j=1}^{n} \exp(y_i)} \qquad (5)$$

Step(1) The architecture of the DNN is designed with three input layers, two hidden layers and one output layer. Activation functions used in DNN is softmax.

Step(2) Initial weights and biases of the DNN are selected with random values.
Step(3) Datasets are constructed and trained using ALO based DNN.
Step(4) ALO parameters such as the population size, maximum iterations, and control parameters are selected
Step(5) Fitness function that evaluates the performance of the DNN.
Step(6) ALO algorithm is utilized to optimize the DNN's hidden neurons. During each iteration of the ALO algorithm, the fitness function is evaluated for each antlion, and the best antlion is selected.
Step(7) DNN's hidden neurons are updated using the information obtained from the ALO algorithm. The best antlion's position is used to update the DNN, improving its performance.
Step(8) ALO-based hidden neuron updation process for multiple iterations are repeated until convergence, where the DNN's performance reaches a satisfactory level.
Step(9) After training, DNN is verified on a separate validation dataset to assess its generalization performance.
Step(10) Finally, trained DNN is tested on a test dataset to evaluate its performance in real-world scenarios.

6. Results and Discussion

The efficacy of the Deep Neural Network classifier, which incorporates Antlion optimization, is established through the evaluation of performance metrics such as precision, recall, F-Score, and specificity. These metrics are computed based on the counts of true positives, true negatives, false positives, and false negatives. A true positive signifies an instance where the model accurately predicts the positive class, while a true negative denotes an instance where the model correctly predicts the negative class. Conversely, a false positive refers to a situation where the model erroneously predicts the positive class, and a false negative pertains to a circumstance where the model incorrectly predicts the negative class. The classification performance metrics are presented in Table 2. Also the classification accuracy tabulated in Table 3. The graphical representation of classification performance metrices by implementing ALO-DNN classifier are shown in Figure.

Table 2. Classification Performance Metrics.

S. No	Disease	Samples	Precision	Recall	F-Scope	Specificity
			ALO- DNN	ALO- DNN	ALO- DNN	ALO- DNN
1	Bacterial blight	2000	0.88	0.90	0.91	0.90
2	Anthracnose	2000	0.89	0.90	0.91	0.91
3	Cercospora Leaf Spot	2000	0.91	0.90	0.89	0.92
4	AlternariaAlternata	2000	0.91	0.88	0.91	0.89
5	Healthy Leaf	2000	0.92	0.89	0.92	0.91

Table 3. Classification Accuracy.

S.No	Disease	Samples	ALO-DNN classifier	ALO-DNN classifier Accuracy (%)
1.	Bacterial blight	2000	1871	93.55
2.	Anthracnose	2000	1868	93.4
3.	Cercospora Leaf Spot	2000	1874	93.7
4.	AlternariaAlternata	2000	1861	93.05
5	Healthy Leaf	2000	1863	93.15
Total		10000	9337	93.37

7. Conclusion

This paper presents an automated approach for disease detection and classification in cotton plants. The system employs FRCM clustering for segmentation and utilizes ALO-DNN for classification. The experimental results demonstrate the remarkable effectiveness of the ALO-DNN classifier, achieving an impressive accuracy of 93.37%. Currently, the system operates on a single digitized color image of a leaf as its input, but there is potential for future expansion. Subsequently, the system may be enhanced to process batches of images, encompassing all parts of a plant, thereby enhancing the accuracy and predictive capabilities of the system.

8. Reference

[1] Gonzalez, R. C. (2009). Digital image processing. Pearson education india.

[2] Sonka, M., Hlavac, V., Boyle, R., Sonka, M., Hlavac, V., & Boyle, R. (1993). Image pre-processing. Image processing, analysis and machine vision, 56-111.

[3] Kaur, G., & Kaur, R. (2012). Image de-noising using wavelet transform and various filters. International Journal of Research in Computer Science, 2(2), 15-21.

[4] Van Veldhuizen, L., De Zeeuw, H., & Van Veldhuizen, L. (1998). Developing technology with farmers. Zed.

[5] Nyvall, R. F., &Nyvall, R. F. (1989). Diseases of Cotton: Gossypium barbadense L. and G. hirsutum L. Field Crop Diseases Handbook, 171-210.

[6] Singh, P., Verma, A., & Alex, J. S. R. (2021). Disease and pest infection detection in coconut tree through deep learning techniques. Computers and electronics in agriculture, 182, 105986.

[7] Alzubaidi, L., Zhang, J., Humaidi, A. J., Al-Dujaili, A., Duan, Y., Al-Shamma, O., ...& Farhan, L. (2021). Review of deep learning: Concepts, CNN architectures, challenges, applications, future directions. Journal of big Data, 8, 1-74.

[8] Vidal, M., & Amigo, J. M. (2012). Pre-processing of hyperspectral images. Essential steps before image analysis. Chemometrics and Intelligent Laboratory Systems, 117, 138-148.

[9] Sivaraman, E., &Arulselvi, S. (2009, June). Gustafson-kessel (GK) clustering approach of TS fuzzy model for nonlinear processes. In 2009 Chinese Control and Decision Conference (pp. 791-796). IEEE.

[10] Tiwari, M., & Gupta, B. (2015). Image denoising using spatial gradient based bilateral filter and minimum mean square error filtering. Procedia Computer Science, 54, 638-645.

[11] Liang, J., Wang, J., & Qian, Y. (2009). A new measure of uncertainty based on knowledge granulation for rough sets. Information Sciences, 179(4), 458-470.

[12] Maji, P., & Roy, S. (2015). Rough-fuzzy clustering and unsupervised feature selection for wavelet based MR image segmentation. PLoS One, 10(4), e0123677.

[13] Abualigah, L., Shehab, M., Alshinwan, M., Mirjalili, S., & Elaziz, M. A. (2021). Ant lion optimizer: a comprehensive survey of its variants and applications. Archives of Computational Methods in Engineering, 28, 1397-1416.

[14] Heidari, A. A., Faris, H., Mirjalili, S., Aljarah, I., &Mafarja, M. (2020). Ant lion optimizer: theory, literature review, and application in multi-layer perceptron neural networks. Nature-Inspired Optimizers: Theories, Literature Reviews and Applications, 23-46.

[15] Wang, H., Li, G., Ma, Z., & Li, X. (2012, October). Image recognition of plant diseases based on backpropagation networks. In 2012 5th International Congress on Image and Signal Processing (pp. 894-900). IEEE.