Improving the performance of leaf disease detection and classification using beetle swarm optimization technique

Penugonda Seetha Rama Krishna, S. Nagarajan
Department of Computer Science and Engineering, Annamalai University, Chidambaram, India

Article Info

Article history:

Received Nov 27, 2024
Revised Apr 11, 2026
Accepted May 17, 2026

Keywords:

Beetle swarm optimization
Color-based segmentation
Convolutional neural network
Feature extraction
Leaf disease detection
Logistic regression

ABSTRACT

The timely identification and diagnosis of leaf diseases is crucial for crop productivity and health. This study proposes a robust approach to this issue by combining beetle swarm optimization (BSO) with other ML models. Four different datasets were used to train our model: apple leaf, grape leaf, plant village leaf, and tomato leaf for disease detection. The process begins with preparing the leaf images, involving contrast enhancement and noise reduction. Through color-based segmentation, we can distinguish healthy regions from diseased ones, aiding in the classification process. Our research demonstrates the effectiveness of the BSO-convolutional neural networks (CNN) method in recognizing and categorizing plant diseases with high accuracy rates. Leveraging the power of BSO to adjust the model's parameters and incorporating color-based segmentation enhances the model's robustness and accuracy. The results of this study highlight the potential of automated disease management systems for agriculture, providing agronomists and farmers with the necessary tools to address and monitor emerging threats to their crops effectively.

Corresponding Author:

Penugonda Seetha Rama Krishna
Department of Computer Science and Engineering, Annamalai University
Chidambaram, Tamil Nadu, India
Email: psrkrishna04@gmail.com

1. INTRODUCTION

Agriculture is a cornerstone of India's economy, yet plant diseases significantly hinder crop growth and yield [1]. These diseases can affect various parts of plants, such as leaves, flowers, roots, and fruits, complicating accurate diagnosis due to the vast number of cultivated plants and the complexity of crops [2]. Pathologists often struggle to diagnose these diseases accurately, leading to the loss of vital qualitative and quantitative crop information [3]. Additionally, many farmers lack the knowledge to detect plant diseases, and relying on the naked eye for identification is time-consuming and unreliable [4], [5].

Automated identification of plant diseases can mitigate these challenges by reducing human effort and providing reliable results. Such systems are particularly beneficial for farmers with limited knowledge of plant diseases. Machine learning (ML) and deep learning (DL) are promising technologies in this field, offering precise and cost-effective solutions for plant disease detection and health monitoring [6]. These technologies enable the accurate identification of plants, the detection of pests and diseases, and the monitoring of nutritional requirements [7].

Early and precise identification of plant diseases is critical for crops to remain healthy. Early detection reduces the need for chemical treatments and helps prevent crop loss [8]. Traditional methods of disease detection can be laborious and require highly skilled personnel. However, advancements in deep learning and computer vision have significantly improved the accuracy of plant disease classification and detection [9]. Recent research has demonstrated the effectiveness of deep learning algorithms in diagnosing plant diseases. For instance, convolutional neural networks (CNNs) have been used to identify multiple plant diseases from extensive image datasets, achieving high accuracy rates [10], [11]. Moreover, optimization techniques such as particle swarm optimization (PSO) and beetle swarm optimization (BSO) have been employed to enhance the performance of these models [12], [13]. Techniques like color slicing and median filtering improve detection accuracy in crops like paddy and soybean [14], [15].

The proposed research aims to develop a more effective method for classifying plant diseases by combining deep learning and optimization techniques. By analyzing extensive collections of annotated leaf data, the methodology seeks to improve classification accuracy by integrating advanced image processing and optimization methods [16], [17]. Additionally, the research explores the potential of incorporating IoT devices and mobile platforms for real-time monitoring and response to plant diseases [18].

This research has the potential to revolutionize how plant diseases are managed and diagnosed, aiding farmers in making informed decisions and improving agricultural productivity [19]. CNN was widely used for this purpose [20]–[22]. The proposed methodology could serve as a foundation for developing new precision agriculture technologies, promoting sustainable practices, and meeting the increasing global food demand. The detailed method will be discussed in the following sections, encompassing data collection, preprocessing, model creation, and optimization methods.

2. MATERIALS AND METHODS

Several procedures detect plant leaf diseases, including segmentation, pre-processing, feature extraction, feature selection, and classification. The proposed BSO method, shown in Figure 1, can help improve these processes.

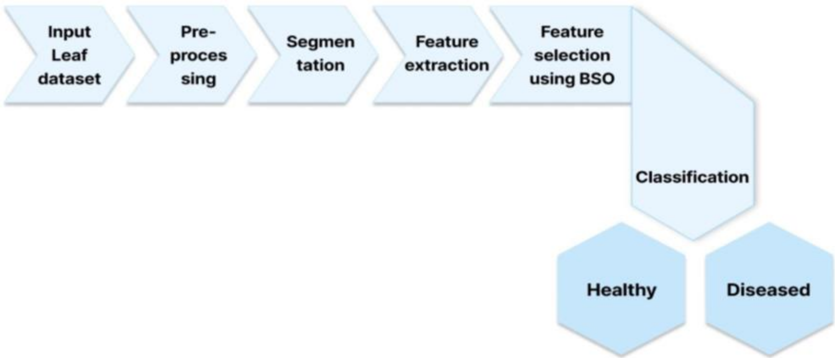

Figure 1. Schematic illustration of the proposed methodology

2.1. Datasets

Some of the datasets featured here contain images of plant leaves that are infected or healthy. These are used in the detection of plant diseases and agricultural research. The PlantVillage dataset [23] is regarded as one of the most prominent collections of plant images in this field. It includes both healthy and diseased specimens. There are varying versions of this repository. The tomato disease detection dataset [24] features images of plants with various pests and diseases. It also includes healthy ones. Leaf disease detection and categorization may be done with Gaussian filters. The grape leaf diseases dataset [25] is also available, as is the tomato dataset. It features pictures of both healthy and sick leaves. Images of unhealthy specimens, pests, and diseases affecting the apple tree leaves may be found in the apple leaf dataset [26]. In that order, the example photos are displayed in Figure 2.

2.2. Pre-processing

Gaussian filters [27], [28] can classify and detect leaf diseases. They are accommodating when dealing with textured or noisy images. The 2D Gaussian filter's equation can be found in (1). The value of the Gaussian filter is expressed as Y(a, b).

$$Y(a,b) = \frac{1}{2\pi\sigma^2} e^{\frac{-a^2+b^2}{2\sigma^2}} \quad (1)$$

Figure 2 illustrates the image datasets employed in this study. Specifically, Figure 2(a) presents a sample from the PlantVillage image dataset, followed by the tomato leaf image data set in Figure 2(b). Furthermore, Figures 2(c) and (d) display samples from the grape leaf and apple leaf image datasets, respectively.

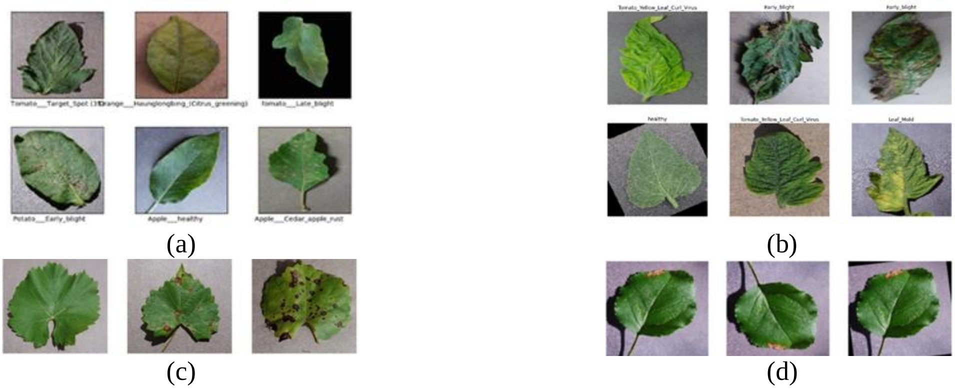

Figure 2. The example photos; (a) PlantVillage image dataset, (b) tomato leaf image dataset, (c) grape leaf image dataset, and (d) apple leaf image dataset

The sigma is the distribution's standard deviation, which dictates the filter's spread. A more considerable value can produce a more robust smoothing or broader distribution. The Gaussian filter kernel can apply a convolution to an input image. The (2) shows the procedure for doing so.

$$J_{smt}(a,b) = \sum_{k=-\infty}^{\infty}\sum_{l=-\infty}^{\infty} Y(k,l).J(a-k,b-l) \quad (2)$$

The resulting image, J_{smt}, is smoothed after applying the Gaussian filter. The pixel value in the input image is represented by J (a-k, b-l), while the value of the Gaussian filter is defined by Y (a, b). The output image is smoothed using a convolution procedure, which minimizes noise and maintains the overall structure's integrity.

2.3. Segmentation using BSO

Here is the protocol for the segmentation. First, initialize a population of beetles. Each beetle represents a potential solution to the optimization problems. Beetle j (B_j) has a position (y_j) in the solution space. Then, define the objective function. The Davies-Bouldin index determines the average level of similarity between clusters. It can then be used to identify those with the worst clustering quality.

$$DB(C_j) = max(R_{jk}), for\ j \neq k \quad (3)$$

Where R_{jk} is the average distance between cluster j (C_j) and cluster k, a particle's overall fitness can be calculated as the mean Davies-Bouldin index over all clusters: Fitness=(1/L)×ΣDB (cluster) for all clusters in the image. In BSO, the beetles explore the solution space to reach the next level. Their movement is influenced by the positions of other insects and their current position. Update the position of each beetle using an equation that combines individual exploration and cooperation among beetles, as shown in (4) and (5).

$$y_j(w+1) = y_j(w) + \alpha \times \Delta y_j \quad (4)$$

Where $y_j(w)$ is the current position of beetle I, α is a step size, and Δ_{yj} is a random exploration vector.

$$y_j(w+1) = y_j(w) + \beta \times \Delta h \quad (5)$$

Where β is a cooperation factor, and Δh is a vector pointing towards a global best solution found by the entire population. Then, we calculate the fitness (objective function value) of each beetle's new position using the objective function: $f_{(w+1)} = f(y_{j(w+1)})$. Further, we keep track of the best solution obtained by any beetle in the population: best = again (f_j). Until the termination condition is satisfied, repeat steps 3-5. The exploration and cooperation parameters (α and β) are adjusted depending on the specific problem and performance requirements. The critical idea in BSO is that beetles explore the solution space individually while benefiting from the best solutions found collectively by the swarm.

2.4. Feature extraction

Color features [29] are visual elements extracted from an image to represent information about the color composition within the picture. These features are widely used in computer vision and image analysis applications, including tasks like categorizing and detecting leaf diseases. By analyzing color characteristics, valuable insights can be gained about the arrangement and balance of colors in an image.

One common color feature is the color histogram, which shows the distribution of colors in an image. This histogram is typically represented as a 3-channel color histogram, where histograms are created for the red, green, and blue channels individually. For each channel, the histogram is constructed by counting the number of pixels with a given color value. Statistical color moments, such as the mean, standard deviation, skewness, and kurtosis, can be computed for each channel to describe color distribution more quantitatively. For example, the mean value of the red channel is calculated by averaging the color values of all pixels in that channel. Another important color feature is the color co-occurrence matrix (CCM), which represents the frequency of color pairs appearing at specific angles and distances. The CCM coordinates indicate how often color i occurs next to color j at a given angle and distance. Color entropy measures the level of disorder or uncertainty in the distribution of colors within an image and is expressed as the probability of a particular color appearing in the frame.

Dominant colors in an image can be identified by calculating their frequency in the color histogram. Additionally, the layout descriptor captures the spatial arrangement of colors, describing their patterns and relationships. When merging texture and color features, statistical methods can be used to integrate these elements. For example, the mean texture feature of the red channel might be computed alongside color features to provide a more comprehensive analysis.

2.5. Feature selection using BSO

When identifying leaf diseases, utilizing BSO involves selecting the most accurate segment of the leaves' color features to help determine the presence of the disease. Here, we explain the steps involved in this process. (a) Extract a set of leaf color features using input images. These can include data from various channels, color moments, and histograms. (b) Establish the proper settings for the BSO method, including the population size, maximum number of iterations, convergence criterion, and more. (c) An objective function should be defined to measure the quality of a subset of the leaf color features. It should be able to tell which features are most accurate in distinguishing healthy leaves from diseased ones. (d) First, establish a population of beetles with binary vectors representing the exclusion or inclusion of certain color features. The group should then be divided into different subsets. (e) The values of the objective function for each subset of the population should be calculated. Then, the classifier will be trained on the selected features and determine if they can help differentiate healthy leaves from diseased leaves. (f) Based on the objective function values, the positions of the beetles should be updated. Beetles that belong to subsets that have better classification capabilities should be favored. (g) The number of iterations achieved or the convergence of the BSO algorithm should be checked. (h) If the convergence or maximum number of iterations has been reached, proceed with the next step. (i) The optimized feature subset derived from the beetle should be obtained using the best binary vector. It should represent the areas of the leaves with the most accurate discrimination when identifying leaf ailments. (j) The classifier should only train on the specified subset of the color features. (k) Cross-validate the classifier's results on a validated dataset or evaluate its performance using the optimized subset. It should also measure its accuracy, recall, precision, and F1 score. (l) After collecting the evaluation results, the selected features should be finalized to provide the best possible performance when detecting leaf diseases.

2.6. CNN architecture

The ResNet architecture, designed for deep neural network training, uses residual blocks to address the vanishing gradient issue, facilitating the training of intense networks [30]. The ResNet-50 model is employed for leaf image classification with a 256×256-pixel input size and outputs two classes: healthy and disease. The model starts with an input layer for leaf images with dimensions 256×256 pixels and three-color channels (RGB). The initial convolutional layer uses a 7×7-pixel kernel with 64 filters, producing 128×128×64 feature maps through padding and stride. Following this, a max-pooling layer reduces the spatial dimensions to 64×64×64 by considering the maximum values in a region. The network then progresses through multiple residual blocks, each containing its filters and skip connections to aid in learning complex features. As the network deepens, the feature maps' dimensions change, representing the feature channels' height, width, and spatial dimensions. Fully connected dense layers follow the residual and convolutional layers, leading to the final output layer, which consists of two units for binary classification. The SoftMax activation function converts these outputs into class probabilities.

3. RESULTS AND DISCUSSION

The performance of various optimization methods, such as PSO, cuckoo search (CS), genetic algorithm (GA), and BSO, in optimizing a specific parameter are compared in Table 1. The best value is achieved across multiple iterations or runs of the algorithm. The BSO method had the best optimization result, with an overall score of 0.83. On the other hand, the worst score was recorded by BSO with a result of 0.26. The mean value indicates the central tendency of the achieved results. The PSO method has a mean value of 0.33. CS, on the other hand, BSO has a mean value of 0.59. The median value measures the central tendency, showing the optimization's results at the midpoint. The data collected by these statistics provide an overview of the performance of various optimization techniques when optimizing a specific parameter. The best and worst scores are displayed because of the highest and lowest achieved values, respectively. The "median" and "mean" values represent the central tendency of the results; the higher the score, the more favorable it is.

Table 1. Performance comparison of optimization techniques for parameter optimization

Parameter	PSO [31]	CS [32]	GA [33]	BSO
Best	0.78	0.79	0.82	0.83
Worst	0.0016	0.0012	0.25	0.26
Mean	0.33	0.39	0.42	0.59
Median	0.39	0.48	0.52	0.60

The presented method was tested by collecting plant leaf images. The collected data were then analyzed using a BSO technique, and the process was evaluated. Table 2 shows performance indicators for the machine learning model (LR), which combine logistic regression with multiple optimization techniques, such as the PSO. These assess the models' effectiveness and quality in specific classification tasks.

Table 2. Performance indicators for various machine learning models

Performance measures	PSO + LR	CS + LR	GA + LR	BSO (proposed technique) +LR
Accuracy	0.81	0.82	0.78	0.84
Sensitivity	0.83	0.92	0.79	0.85
Precision	0.77	0.82	0.86	0.89
Dice coefficient	0.87	0.8	0.75	0.85
Jaccard	0.78	0.76	0.92	0.84

The proportion of correctly classified instances measures the model's accuracy. It helps to evaluate the model's correctness when making predictions on both negative and positive cases. PSO+LR, CS+LR, GA+LR, and BSO+LR all achieved an accuracy of at least 0.81. Among the models, the BSO+LR model is the most accurate at 0.84. The sensitivity of a model is computed by considering the likelihood of it identifying true positives from all known instances. It can then be used to measure its effectiveness in capturing such cases. The sensitivity of the PSO+LR model was set at 0.83. Other models that have better sensitivity include the CS+LR, LR+GA, and BSO+LR.

The precision ratio serves as a gauge for the model's positive prediction accuracy. It indicates how capable the model is of making accurate predictions. The PSO+LR, CS+LR, GA+LR, and BSO+LR made accurate predictions with 0.77, 0.82, 0.86, and 0.89 precision, respectively. The concept of the dice coefficient measures the overlap of the predicted and actual positive instances in each set. It is commonly used in the classification of binary types and image segmentation. The PSO+LR achieved a dice coefficient of 0.87, while CS+LR had a value of 0.80. On the other hand, the GA+LR had a value of 0.75, and BSO+LR achieved a value of 0.85.

The classification tool known as the Jaccard Index considers the similarity between predicted and actual instances when performing segmentation or classification tasks. The classification tools that performed well were the PSO+LR, CS+LR, GA+LR, and BSO+LR. The highest score was achieved by the BSO+LR, which had a value of 0.84. The various metrics analyzed in this study provide a detailed analysis of the capabilities of the models when it comes to capturing positive instances and accuracy. The results indicate that BSO+LR performed well regarding sensitivity, precision, and the Dice coefficient. It is essential to consider the classification task's specific requirements when picking a suitable model.

Table 3 shows the performance indicators that apply to the different optimization algorithms: PSO, CS, BSO, and GA. These are used in combination with a CNN for specific applications or tasks. These include sensitivity, precision, Jaccard index, and dice coefficient. The PSO-CNN combination has an accuracy rate of 83%, CS-CNN has an accuracy rate of 84%, BSO-CNN has an accuracy rate of 91%, and GA-CNN has an accuracy rate of 85%. The sensitivity of PSO+CNN, CS+CNN, BSO+CNN, and GA+CNN is 86%, 85%, and 93%, respectively. On the other hand, these two groups show the highest sensitivity values at 93%, indicating they are better at identifying positive cases.

Table 3. Performance indicators for various deep learning models

Performance measures	PSO+CNN	CS+CNN	GA+CNN	BSO+CNN (proposed technique)
Accuracy	0.83	0.84	0.85	0.91
Sensitivity	0.86	0.93	0.85	0.93
Precision	0.82	0.84	0.88	0.93
Dice coefficient	0.91	0.85	0.86	0.95
Jaccard	0.88	0.81	0.86	0.95

The dice coefficient measures the overlap between positive areas and the predictions. The highest score is recorded by CNN+BSO, which has a 95% correlation. The positive predictive value of PSO+CNN is 82%, CS+CNN is 84%, GA+CNN is 88%, BSO+CNN is 93%, and so on. The combination with the highest precision, BSO+CNN, makes fewer errors than the others. The Jaccard index is the classification-over measure of the correlation between predicted and actual positive regions. The CNN+BSO index has the highest score at almost one hundred percent, which indicates a substantial overlap between the predicted and the actual positive areas. As seen in Figure 3, BSO+CNN outperforms the other groups on a wide range of measures.

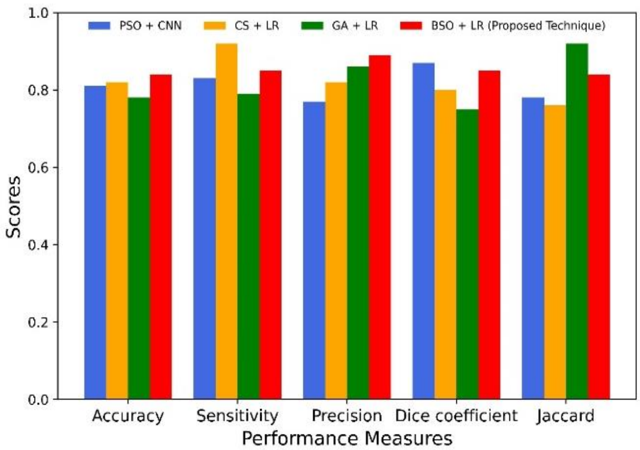

Figure 3. The performance of deep learning techniques across various metrics

4. CONCLUSION

The timely diagnosis and identification of diseases in leaves is a critical factor in maintaining the health and productivity of crops. This research suggests a unique automated plant disease identification and classification method that combines CNN and BSO. The research utilizes four distinct datasets to train and assess our model. These include the PlantVillage, apple leaf, and grape leaf datasets. The study provides a novel and comprehensive solution to the issue of identifying and managing leaf diseases in agriculture. By integrating BSO with CNN and employing advanced image processing methods, we offer a potent tool to help farmers and agronomists improve crop productivity and health. As the evolution of automated systems continues, they have the potential to make a significant contribution to global food security and sustainability efforts.

ACKNOWLEDGMENTS

The authors acknowledge the support of the Department of Computer Science Engineering, Annamalai University, Tamil Nadu, India, to conduct this research.

FUNDING INFORMATION

Authors state there is no funding involved.

AUTHOR CONTRIBUTIONS STATEMENT

This journal uses the Contributor Roles Taxonomy (CRediT) to recognize individual author contributions, reduce authorship disputes, and facilitate collaboration.

Name of Author	C	M	So	Va	Fo	I	R	D	O	E	Vi	Su	P	Fu
Penugonda Seetha Rama Krishna	✓	✓	✓	✓	✓	✓		✓	✓	✓				
S. Nagarajan	✓	✓				✓		✓	✓	✓	✓	✓	✓	

C : **C**onceptualization
M : **M**ethodology
So : **So**ftware
Va : **Va**lidation
Fo : **Fo**rmal analysis
I : **I**nvestigation
R : **R**esources
D : **D**ata Curation
O : Writing - **O**riginal Draft
E : Writing - Review & **E**diting
Vi : **Vi**sualization
Su : **Su**pervision
P : **P**roject administration
Fu : **Fu**nding acquisition

CONFLICT OF INTEREST STATEMENT

The authors declare that they have no conflict of interest.

INFORMED CONSENT

Informed consent was obtained from all individual participants involved in the study.

ETHICAL APPROVAL

This study did not involve any experiments with human participants or animals, and therefore ethical approval was not required.

DATA AVAILABILITY

The data that support the findings of this study are available from the corresponding author upon reasonable request.

REFERENCES

[1] S. S. Chouhan, A. Kaul, U. P. Singh, and S. Jain, "Bacterial foraging optimization based radial basis function neural network (BRBFNN) for identification and classification of plant leaf diseases: an automatic approach towards plant pathology," *IEEE Access*, vol. 6, pp. 8852–8863, 2023, doi: 10.1109/ACCESS.2018.2800685.

[2] K. P. Ferentinos, "Deep learning models for plant disease detection and diagnosis," *Computers and Electronics in Agriculture*, vol. 145, pp. 311–318, 2024, doi: 10.1016/j.compag.2018.01.009.

[3] A. A. Bharate and M. Shirdhonkar, "A review on plant disease detection using image processing," in *Proceedings 2021 International Conference on Intelligent Sustainable Systems (ICISS)*, Palladam, India, Dec. 7–8, 2021, pp. 103–109, doi: 10.1109/ISS1.2017.8389326.

[4] B. Tlhobogang and M. Wannous, "Design of plant disease detection system: a transfer learning approach work in progress," in *Proceedings 2022 IEEE International Conference on Applied System Innovation (ICASI)*, Chiba, Japan, Apr. 13–17, 2022, pp. 158–161, doi: 10.1109/ICASI.2018.8394556.

[5] A. M. Mutka and R. S. Bart, "Image-based phenotyping of plant disease symptoms," *Frontiers in Plant Science*, vol. 5, p. 734, 2021, doi:10.3389/fpls.2014.00734.

[6] R. Tijare, P. Khade, and R. Jain, "The survey of disease identification of cotton leaf," *International Journal of Innovative Research in Computer and Communication Engineering*, vol. 3, no. 11, 2015, doi: 10.15680/IJIRCCE.2015. 0311131.

[7] Y. M. A. Algani, O. J. M. Caro, L. M. R. Bravo, C. Kaur, M. S. Al Ansari, and B. K. Bala, "Leaf disease identification and classification using optimized deep learning," *Measurement: Sensors*, vol. 25, p. 100643, 2023, doi: 10.1016/j.measen.2022.100643.

[8] K. Uddin, M. H. Khan, M. Rahman, and A. Fahim, "Automated identification of plant disease using deep learning," *Journal of Biological Sciences*, vol. 2, pp. 48-56, 2022, doi: 10.29329/JofBS.2022.445.02.

[9] S. V. Militante, B. D. Gerardo, and N. V. Dionisio, "Plant leaf detection and disease recognition using deep learning," in *Proceedings 2023 IEEE Eurasia Conference on IoT, Communication and Engineering (ECICE)*, 2023, pp. 579–582, doi: 10.1109/ECICE47484.2019.8942686.

[10] S. Abbas, M. Jain, S. Gour, and S. Vankudothu, "Tomato plant disease detection using transfer learning with C-GAN synthetic images," *Computers and Electronics in Agriculture*, vol. 187, p. 106279, 2021, doi: 10.1016/j.compag.2021.106279.

[11] M. A. Jan, S. Khamis, F. Mohammad, and H. J. Fariman, "Feature decision-making ant colony optimization system for an automated recognition of plant species," *Expert Systems with Applications*, vol. 42, no. 5, 2023, doi: 10.1016/j.eswa.2014.11.011.

[12] R. Patil, S. Udgave, S. More, D. Nemishte, and M. Kasture, "Grape leaf disease detection using k-means clustering algorithm," *International Research Journal of Engineering and Technology (IRJET)*, vol. 3, no. 4, pp. 2330–2333, 2024.

[13] M. Kumar, P. Gupta, P. Madhav, and Sachin, "Disease detection in coffee plants using convolutional neural network," in *Proceedings 2024 5th International Conference on Communication and Electronics Systems (ICCES)*, Coimbatore, India, Jun. 2024, pp. 755–760, doi: 10.1109/ICCES48766.2020.9138000.

[14] A. Singh and M. L. Singh, "Automated blast disease detection from paddy plant leaf—a color slicing approach," in *Proceedings 2023 7th International Conference on Industrial Technology and Management (ICITM)*, Oxford, UK, Mar. 7–9, 2023, pp. 339–344, doi: 10.1109/ICITM.2018.8333972.
[15] Z. B. Husin, A. Y. B. M. Shakaff, A. H. B. A. Aziz, and R. B. S. M. Farook, "Feasibility study on plant chili disease detection using image processing techniques," in *Proceedings 2022 Third International Conference on Intelligent Systems Modelling and Simulation*, Kota Kinabalu, Malaysia, Feb. 8–10, 2022, pp. 291–296, doi: 10.1109/ISMS.2012.33.
[16] L. Ma, J. Fang, Y. Chen, and S. Gong, "Color analysis of leaf images of deficiencies and excess nitrogen content in soybean leaves," in *Proceedings 2023 International Conference on E-Product E-Service and E-Entertainment*, Henan, China, Nov. 7–9, 2023, vol. 11541023, pp. 1–3, doi: 10.1109/ICEEE.2010.5661235.
[17] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, 3rd ed. Upper Saddle River, NJ, USA: Pearson Prentice Hall, 2022.
[18] J. Kittler and J. Illingworth, "Minimum error thresholding," *Pattern Recognition*, vol. 19, pp. 41–47, 1986, doi: 10.1016/0031-3203(86)90030-0.
[19] ASM International, *Practical Guide to Image Analysis*. Novelty, OH, USA: ASM International, 2000.
[20] S. Mohanty *et al.*, "Using deep learning for image-based plant disease detection," *Frontiers in Plant Science*, vol. 7, p. 215232, 2024, doi: 10.3389/fpls.2016.01419.
[21] Brahimi *et al.*, "Deep learning for plant diseases: detection and saliency map visualization," *Human and Machine Learning: Visible, Explainable, Trustworthy and Transparent*, pp. 93–117, 2023, doi: 10.1007/978-3-319-90403-0_6.
[22] P. Remagnino, S. Mayo, P. Wilkin, J. Cope, and D. Kirkup, "Machine learning for plant leaf analysis," *Computational Botany: Methods for Automated Species Identification*, pp. 57–79, 2024, doi: 10.1007/978-3-662-53745-9_4.
[23] D. Hughes and M. Salathé, "An open-access repository of images on plant health to enable the development of mobile disease diagnostics," *arXiv preprint arXiv:1511.08060*, 2023.
[24] M. Gehlot, R. K. Saxena, and G. C. Gandhi, "Tomato-Village: a dataset for end-to-end tomato disease detection in a real-world environment," *Multimedia System*, vol. 29, no. 6, pp. 3305-3328, 2023, doi: 10.1007/s00530-023-01158-y.
[25] P. Jiang, Y. Chen, B. Liu, D. He, and C. Liang, "Real-time detection of apple leaf diseases using deep learning approach based on improved convolutional neural networks," *IEEE Access*, vol. 7, pp. 59069–59080, 2024, doi: 10.1109/ACCESS.2019.2914929.
[26] B. Liu, Y. Zhang, D. He, and Y. Li, "Identification of apple leaf diseases based on deep convolutional neural networks," *Symmetry*, vol. 10, no. 1, 2024, doi: 10.3390/sym10010011.
[27] M. Wang, S. Zheng, X. Li, and X. Qin, "A new image denoising method based on gaussian filter," in *Proceedings 2024 International Conference on information science, electronics and electrical engineering*, Sapporo, Japan, pp. 163-167, doi: 10.1109/InfoSEEE.2014.6948089.
[28] A. Katukuri and R. Boda, "A multi-objective randomly updated beetle swarm and multi-verse optimization for brain tumor segmentation and classification," *Computer Journal*, vol. 65, no. 4, pp. 1029–1052, 2022, doi: 10.1093/comjnl/bxab171.
[29] J. Chaki and N. Dey, *Image Color Feature Extraction Techniques: Fundamentals and Applications*, 2020.
[30] L. Zhang, Y. Bian, P. Jiang, and F. Zhang, "A transfer residual neural network based on ResNet-50 for detection of steel surface defects," *Applied Sciences*, vol. 13, no. 9, p. 5260, 2023, doi: 10.3390/app13095260.
[31] A. G. Gad, "Particle swarm optimization algorithm and its applications: a systematic review," *Archives of Computational Methods in Engineering*, vol. 29, pp. 2531–2561, 2022, doi: 10.1007/S11831-021-09694-4.
[32] A. S. Joshi, O. Kulkarni, G. M. Kakandikar, and V. M. Nandedkar, "Cuckoo search optimization—a review," *Materials Today: Proceedings*, vol. 4, no. 8, pp. 7262-7269, 2024, doi: 10.1016/j.matpr.2017.07.055.
[33] S. Katoch, S. S. Chauhan, and V. Kumar, "A review on genetic algorithm: past, present, and future," *Multimedia Tools and Applications*, vol. 80, pp. 8091–8126, 2021, doi: 10.1007/s11042-020-10139-6.

BIOGRAPHIES OF AUTHORS

Penugonda Seetha Rama Krishna is currently working as software engineer in ANZ. He completed his M.Tech. from Sunflower Engineering College, J.N.T.U Kakinada in 2014. He completed his M.Sc. from K.B.N College, Vijayawada, Acharya Nagarjuna University in 2009. He received his B.Sc. from V.R.S and Y.R.N College, Chirala in 2006. He can be contacted at email: psrkrishna04@gmail.com.

Dr. S. Nagarajan is currently working as an associate professor in the Department of Computer Science and Engineering, Government College of Engineering Srirangam, Trichy on deputation. Previously, he worked as an assistant professor in the Department of Computer Science and Engineering, Faculty of Engineering and Technology, Annamalai University from July 2002 to May 2017. He received his B.E. degree in Electronics and Communication Engineering from Bharathidasan University in the year 1994 and M.B.A. (Personnel Management) from Madurai Kamaraj University in the year 2002. He completed both his M.E. degree and Ph.D. in Computer Science and Engineering from Annamalai University in the year 2005 and 2016 respectively. He has 26 years of teaching service to his credit. He has published more than 35 research papers in international journals and conferences. His research interests include computer vision, artificial intelligence, image processing, and pattern recognition. He is a life member in CSI, member in IAENG, CSTA, and UACEE. He is a Fellow of Institution of Engineers (India). He can be contacted at email: drsnagarajan_cse@gces.edu.in.