International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
ISSN:2147-6799 www.ijisae.org Original Research Paper

Medical Plant Identification and Classification Using Average Weight Inertia Based Cat Swarm Optimization and Enhanced Convolutional Neural Network Algorithm

Haorongbam Anand Singh[1], Maibam Maikel Singh[2], Takhellambam Sonamani Singh[3], Th. Rupachandra Singh[4]

Submitted: 10/10/2023 **Revised**: 29/11/2023 **Accepted**: 10/12/2023

Abstract: Medicinal plants have long been studied and taken into consideration because of how crucial they are to maintaining human health. However, finding medicinal plants takes time, is laborious, and needs knowledgeable professionals. Therefore, the vision-based method may aid both scientists and common people in precisely and swiftly identifying herbaceous plants. In this paper, an upgraded convolutional neural network (ACSO+ECNN) and an average weight inertial cat colony optimisation (ACSO) method are suggested to boost classification and recognition skills. health tree. This work's four stages include pre-processes, feature extractions and selections with classifications. An adaptive median filter (AMF) is applied during preprocessing to efficiently manage noise. Following that, texture and colour characteristics are extracted as features. Four-dimensional histograms (QH) are used to extract colour data, while grey level co-occurrence matrices (GLCM) are used to extract texture information. Following that, the ACSO method is used to identify significant and pertinent characteristics from the provided image collection. The ACSO algorithm's best likelihood values are used for this. Finally, the ECNN algorithm is used to conduct recognition and classification. The predictions of the ACSO+ECNN model are combined using ECNN to create a model that accurately identifies the proper manufacturer based on a certain kind and set of features. The findings of the experiment suggest the proposed ACSO+ECNN method provides higher accuracy, precision, recall, and minimum execution time than existing methods.

Keywords: *Medical plant classification, Enhanced Convolutional Neural Network (ECNN) and Average weight inertia based Cat Swarm Optimization (ACSO) algorithm, feature extraction, feature selection and classification*

1. Introduction

Due to their nutritional and therapeutic qualities, medicinal plants like trees, shrubs, and grasses have long been a part of traditional medicine and used for their antioxidant, anti-allergic, anti-inflammatory, and antibacterial properties because of their bioactive constituents including phenolic compounds, carotenoids, anthocyanins, and other bioactive chemicals. Their particular geographic distributions are governed by environmental conditions that have shaped their evolutions over time. According to statistics, 14 to 28% of all plants have medical properties [2]. Furthermore, 3-5% of patients in developed nations, and more than 80% of rural populations in poor countries, and around 85% of people in sub-Saharan Africa utilise medicinal plants to heal disorders brought on by their qualities.

Furthermore, as a result of bad repercussions and modern medication's side effects, people in developed nations have begun to use traditional herbal treatments for disease treatments and controsl. These plants can be utilised not just for medical purposes but also as food, drink, and even cosmetics [3]. Unfortunately, there are a lot of fake, subpar, damaged, or improperly kept herbal medications made and marketed globally, which might be dangerous to consumers.

Roots, flowers, and leaves are just a few of the several organs that plants have. One of the most crucial plant organs, leaves, differ greatly between species and types in terms of their colour, form, and textural qualities. However, sometimes it can be challenging to identify medicinal plants because of how similar their leaves are. Furthermore, due to their similarity and, more importantly, their unpredictability during the growth period, leaves cannot be regarded an acceptable alternative for categorising plants [4].

Mecinal plants can be used for treating human illnesses and ailments. Herbal remedies come in a wide variety of forms and might range from region to region, giving rise to patterns in "size" and "shape" that are comparable [5]. The roots and leaves of these plants are all incredibly therapeutic. Several plants, including Karpooravalli (Coleus ambonicus), Podina (Mentha arvensis), Neem (Adidirachta indica), Thudhuvalai (Solanum trilobatum), and basil (Ocimum sanctum), are used in society today. Some leaves have special medicinal characteristics, such as

[1]*haorongbamanandsingh@gmail.com, CMJ University, Meghalaya*
[2]*maikelmaibam@manipuruniv.ac.in, Department of Computer Science, Manipur University*
[3]*sona.cse2013@gmail.com, Manipur Institute of Technology, Manipur University*
[4]*rupachandrath@gmail.com, Department of Computer Science, Manipur University*

healing dyspepsia, colds, skin disorders, and blood purification.

To extract the most pertinent information from medical databases nowadays, several feature extraction and selection approaches are applied. These selection methods are used to forecast various illnesses using medical data. The majority of feature selection techniques employ evolutionary and heuristic techniques to lessen computing complexity. These techniques can effectively and quickly solve high-dimensional optimisation issues [6] [7]. The most widely used meta-heuristic methods that draw their inspiration from nature are swarm-based algorithms. Swarm intelligence, or SI, is one of the AI-based techniques that gives self-organizing and decentralised systems collective behaviours. It consists of several simple agents that converse just locally and within their surroundings.

Each individual plant must be correctly allocated to a descending list of connected plant groupings as part of the categorization process. The limitations of human classification approaches for recognising medicinal plants are addressed in [8], which presents an automated classification system based on imagegraphs of medicinal plant leaves. Our method starts by preprocessing images of medicinal plant leaves; it next computes ten form features (SF) and five texture features (TF); and last, it uses a support vector machine (SVM) classifier to categorise medicinal plant leaves. The classifier produced an average identification rate of 93.3% when used on 12 distinct images of medicinal plant leaves. The outcomes demonstrated that multi-feature extractions from images of leaves and usage of SVMassisted in categorizing medicinal plants.

Though many studies have categorized plants their accuracies have not produced any notable results. Hence, identifications and classifications of medicinal plants is the primary goal of this research.The identification of various species and the overall effectiveness of current methods are problematic. In order to solve the aforementioned issues, DL (deep learning) approaches have been suggested in this work to improve overall performanceswhere the processes of pre-processing, feature extractions, feature selections, and classifications. Effective algorithms executable on available data sets are used to deliver accurate results.

The rest of this article is divided into the following sections: A survey of the literature on the naming and categorization of medicinal plants is included in Part 2 of the article. The suggested ACSO+ECNN approach to identify and categorise medicinal plants is described in depth in Part 3. The experiment's findings are presented in Section 4. The article is finished in Part 5.

2. Related Work

In [9], De Luna et al. (2017) using artificial neural networks in conjunction with image processing techniques to extract leaf-related information in order to recognise and identify various medicinal plants from the Philippines. Actual specimens of 12 distinct medicinal plant leaves were gathered, and each leaf was imagegraphed separately. Image processes extract required features. The system can recognise the medicinal plant leaf species under test by behaving as an independent brain network using an artificial neural network. The system can also offer details on illnesses that plants can treat. A feature dataset of 600 images was utilised for training, including 50 images for each medicinal plant.

In [10], Anton et al. (2020) used image processing techniques to identify impacted leaf texture using the convolutional neural network (CNN) approach and GLCM extraction. Downy mildews, Septoria leaf spots, bacterial leaf spots, target spots, downy mildews, leaf curls, spider mites, two-spotted spider mites, and leave1 mould are among the diseases that typically affect tomato leaves. Because it is more accurate at recognising and categorising plant illnesses than CNN alone, the GLCM colour moment and CNN approach were used for this study. In this work, we combined CNN techniques with GLCM colour moments.

Gopal et al. (2012) attempted to put such a method into practise utilising image processing and leaf images as the starting point for categorization in [11]. The closest match to the query is returned by the programme. The algorithm was put into practise, and ten different plant species were used to assess the system's efficacy. Each plant has ten leaves, thus the programme learns on 100 leaf images and tests it on 50 leaf images for increased effectiveness of algorithm’s executions.

Pankaja et al. (2020) presented a technique for classifying and identifying leaves in [12] by fusing the whale optimisation algorithm (WOA) with random forest (RF). On the Swedish and Flavia leaf databases, this work was done. Prior to feature extraction, preprocessing initially reduces noises in data or increases quality. To get around the dimensionality issue, WOA is employed. In addition, leaves are identified using RF classifier. In comparison to previous approaches, the method exhibits great accuracy with shorter execution times. Better categorization and identification of leaves used for therapeutic purposes are made possible by this work.

CNN was used by Quoc and colleagues (2020) in [13] contained images of Vietnamese medicinal plants. VGG16, Resnet50, InceptionV3, DenseNet121, Xception, and MobileNet are amongst frameworks evaluated. The highest accuracy of the outlier is 88.26%. One may assume that this approach would significantly aid in the identification and preservation of priceless medicinal plants.

3. Proposed Methodology

This work presents ACSO+ECNN algorithm to detect and categorize medicinal plants using the processess of pre-processes, feature extractions and selections with classifications. The overall architecture of the suggested ACSO+ECNN algorithm is shown as Figure 1.

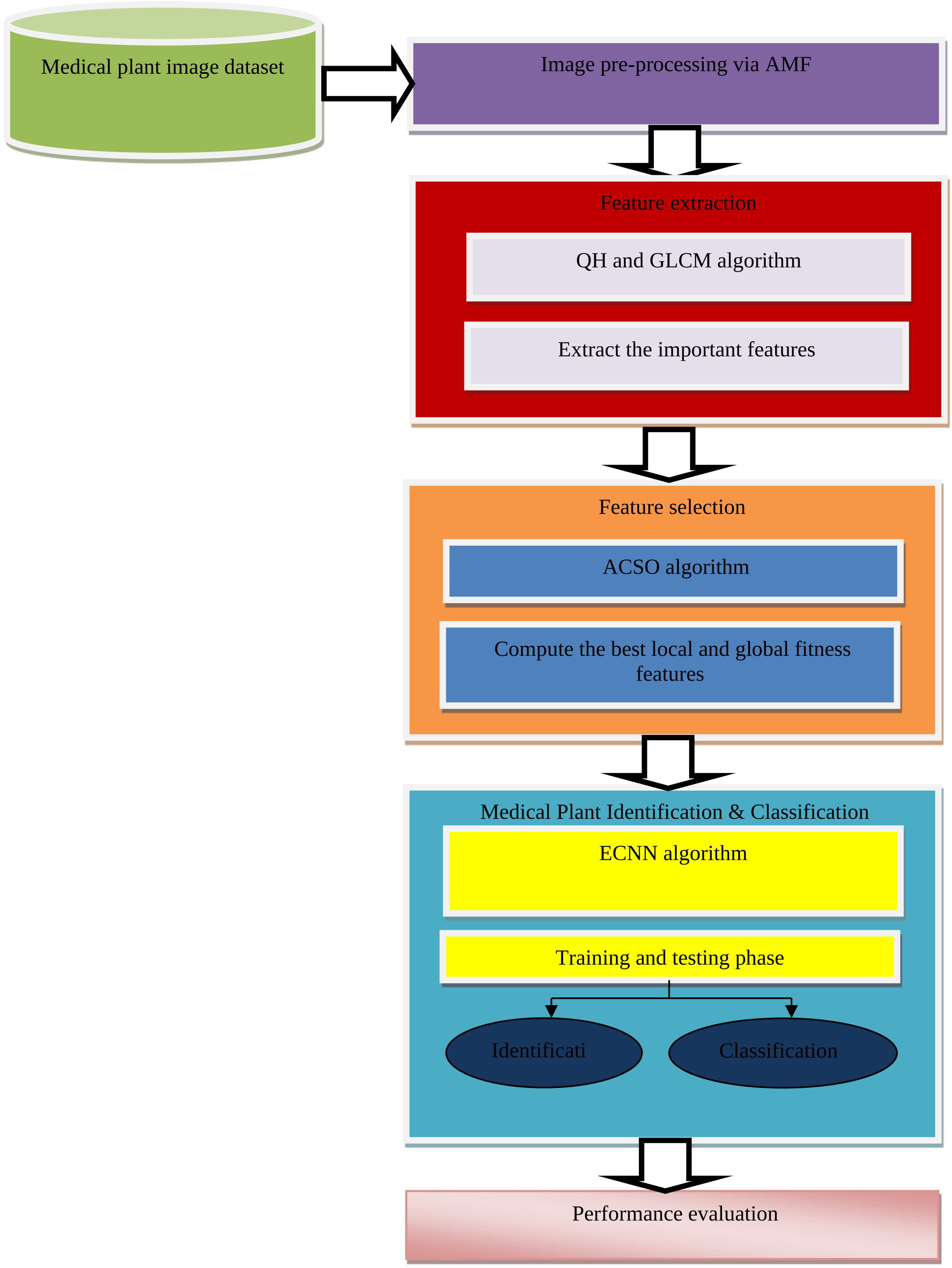

Fig 1 Overall block diagram of the proposed ACSO+ECNN framework

3.1. Image Pre-processing through Adaptive Median Filtering (AMF)

The AMF method is used in this study to do noise reduction, which tries to eliminate unneeded noise from certain imagegraphs. It primarily serves as a model that can be relied on and utilised to better imaging outcomes when there is a lot of noise present. To ascertain which pixels in the image are impacted by noise, AMF conducts spatial processing [14]. By comparing each pixel in the image with its immediate neighbours, MFA categorises pixels as noise. Both the neighborhood's size and the comparison threshold are scalable [15]. Noise is a pixel that is not fundamentally aligned with comparable pixels but differs from most of its neighbours. The average values of neighbouring pixels are passed through noisy labelling tests and are noisy labels are replaced, thus enhancing image qualities and lowering image noises.

In response to the quantity of candidate sounds in a region, the adaptive filter's window size is raised. According to equation (1), the window size in each region is determined by the quantity of noisy pixels there. Extensive simulations were performed to guarantee the optimum outcomes based on a Mean Squared Error (MSE) threshold to enlarge the window dependent on the quantity of noisy pixels in a location. development.

$$W\ (M) = \sum_{p} C(p, D_P) + \sum_{q \epsilon N_p} T\left[\left|M_P - M_q\right| = 1\right] + \sum_{q \epsilon N_p} T\left[\left|M_P - M_q\right| > 1\right] \quad ((1)$$

W represents window matching functions, while M represents their optimal solutions. The true pixel in the provided image is p, and the neighbouring pixel of p is q. $N_P$denotes the collection of pixels next to p. $M_P and M_q$ are the corresponding estimate windows of p and qwhile C implies matching window costs. $T[\cdot]$ represents logical functions that evaluate statements and return values of 1 for true cases. The window sizes in equation (1) are expanded when there are no noise-free pixels in windows or when centres of windows contain noisy pixels. Window sizes used for filtering image pixels are adaptive implying they will be extended if the given requirements are not satisfied. The pixel will be filtered using the window median if the condition is satisfied. Let I_{ij} represent the corrupted image's pixels, I_{min} represents minimum values of pixels, I_{max} represents maximum values of window pixels, W represents current sizes of the windows, W_{max} represents maximum sizes of resultant windows, and I_{med} represents intersection window's medians. This filtering approach algorithm has two layers, as illustrated below.

Level A:

a) If $I_{min} < I_{med} < I_{max}$, then median values are not noises and the model executes Level B for determining noisy pixels.

b) Level A is then repeated until averages are no longer noisyor until maximum window sizes are achieved for the model to change to level B, when the averages are presented as filter's final image pixel values.

Level B:

a) If $I_{min} < I_{ij} < I_{max}$then current pixel values are not noises, and hence filtered image pixels are constant.

b) Else if image pixels are either equal to $I_{max} or I_{min}$(corrupted), subsequently filtered image pixels are allocated with median values from Level A

3.2 Color and texture feature extraction using Quad Histogram (QH) and GLCM

Feature extractions imply defining sets of features for efficient representations of information for analyses and classifications. The colors and textures are extracted to classify medical plants

✓ Color feature extraction is done by using Quad Histogram

✓ Texture feature extraction is done by using GLCM

3.1.1. Color feature extraction by using Quad Histogram (QH)

Based on the four-pole histogram, colour feature extraction is carried out in this research project. The image is divided into uniform chunks of various sizes using quad-tree decomposition.

The histogram may be easily generated by reading each pixel in the image one at a time and incrementing the relevant box in the histogram. The graph counts pixels of each category based on images' block complexities, quad tree decompositions separate them into subsequent quadrants. When sub-images do not form homogeneous blocks, they are split into four equally sized sub-imagess. This split is seen in Figure 2. Sub-images are homogeneous blocks when sums of maximum and lowest values of block components exceed applicable threshold values.

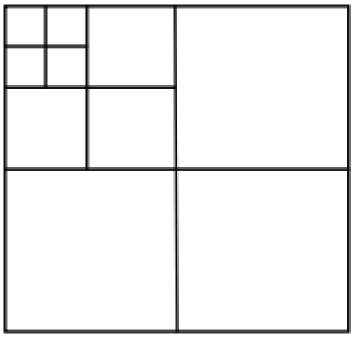

Fig 2 Quad tree subdivision

For RGB images of size M× N × 3, the quad tree decomposition approach employs threshold values of red, green, and blue colour blocks. Quad tree decompositions turn the image first into square images of sizes M× N into $M' \times M'$. M' is of the second order and multiples of n, where n stands for sizes of the smallest homogeneous blocks.

3.1.2. Texture feature extraction using GLCM

An image's texture is a crucial component for highlighting areas of interest. In many structural analysis applications, GLCM has been utilised extensively and is still a crucial feature extraction technique. It is a technique for capturing an image's visual information for indexing and retrieval. In order to specify pertinent data for a certain application's computing job, feature extraction is employed.

GLCM are matrices where counts of rows and columns are equal to counts of gray levels, G, in images. The matrix elements $P\ (i, j \mid \Delta x, \Delta y)$ are relative frequencies with which two pixels, separated by pixel distances $(\Delta x, \Delta y)$, occur within neighborhoods, one with intensities 'i' and other with intensities 'j'. The matrix elements $P\ (i, j \mid d, \theta)$ contain second order statistical probability values for changes between gray levels 'i' and 'j' at particular displacement distances d and at particular angles (ɵ). Using multiple intensity levels G implies storing temporary data, i.e. a $G \times G$ matrices for combinations of $(\Delta x, \Delta y)\ or\ (d, \theta)$. Because of their high dimensionality, GLCMs are sensitive to sizes of structural samples used to estimate them [16]. Hence, grey level counts are frequently reduced. GLCM plots depict connections between reference pixels (i) and neighbouring pixels (j).

A well-known statistical method for obtaining second-order texture data from images is GLCM. The counts of times an image element with value i occurs adjacent to an image element with value j is shown in each (i, j) segment of the GLCM. Typically, an image element with value j displays adjacent to an image with value i for the grey level (grayscale power). Calculations of texture features employ GLCM information to generate measurements of intensity changes at target pixels. The pixel counts and relative distances between pairs of pixels are two criteria that are typically utilised to generate an occurrence matrix. Although many additional combinations are conceivable, quantization is often done in four directions (e.g., 0°, 45°, 90°, and 135°).

The most useful GLCM properties are: second angular momentum (ASM), contrasts, correlations, inverse difference momentums, total entropies, and correlated information measurements. These features look promising.

Energy: Energy (E) is a measurement of how repeating pixel pairings are. It assesses the consistency of a image. The energy value will be high when the pixels are highly comparable. It is constructed as follows:

$$E = \sum_{i,j=0}^{N-1} p_{i,j}^2 \qquad (2)$$

Where, p represents counts of GLCM.

Entropy: Statistical measures of randomness that can be used to characterize input image textures.

$$\text{Entropy} = -\sum\sum p(i,j)\log p(i,j) \qquad (3)$$

Contrasts: Measure local variationss in GLCM and compute intensities between pixels and their neighbors in images. Contrast is 0 for constant images.

$$\text{Contrast} = \sum\sum (i-j)^2\, p(i,j) \qquad (4)$$

Where p(i,j) = pixel at location (i,j)

Correlations: Measures the joint occurance probabilities of specified pixel pairs.

$$\text{Correlation} = \frac{\sum_{i=0}^{G-1}\sum_{j=0}^{G-1}(i-\mu_i)(j-\mu_j)p(i,j)}{\sigma_i\sigma_j} \qquad (5)$$

Homogeneity: these measures assess closeness of element distribution in GLCM based on GLCM diagonals.

$$\text{Homogeneity} = \sum_{i,j}\frac{P(i,j)}{1+|i-j|} \qquad (6)$$

$$\text{Dissimilarity} = \sum_{i,j=0}^{N-1} P_{i,j}\,|i-j| \qquad (7)$$

The first element in equation (7) is for the vertical coordinates, whereas the second element is for the horizontal coordinates.

All of these characteristics offer strong selective strength to separate between two different image kinds. The extracted GLCM second-order textural features are statistically computed. Calculations are made for the second six orders in terms of energy, entropy, homogeneity, correlation, homogeneity, dissimilarity, and contrast. Energy is a metric for how soft a image is. When measuring local-level changes, contrast assigns lower values to images with low contrast and higher values to those with strong contrast. The degree of homogeneity in the GLCM gauges how evenly distributed the components are. It has a range of 0 to 1. For diagonal GLCM, uniformity is 1. Chance is measured by entropy. Effectively, the distinction between two image features is produced by the difference. These values shed further light on the specific sort of cancer—or cancers—into which they might potentially develop. In light of the poor data provided, the GLCM approach is employed to produce more useful characteristics by employing energy, entropy, uniformity, correlation, homogeneity, difference, and contrast texture.

3.2. Feature selection via Average weight inertia based Cat Swarm Optimization (ACSO) algorithm

In order to increase the classification accuracy of medicinal plants, the ACSO algorithm is used in this section to choose the ideal qualities. Based on the typical cat behaviour, the CSO algorithm was created. It has been shown that cats prefer to relax and observe their surroundings over chase items since the latter consumes an excessive amount of energy. The algorithm is split into two sub-modes to reflect these two crucial feline behavioural traits, which CSO refers to as "search mode" and "tracking mode," denoting two distinct algorithmic processes. While search mode simulates cats' behaviour when they are at rest and taking in their surroundings, pursuit mode simulates cats' behaviour when they are running after a target [17].

A collection of solutions are represented by each cat, each of which has a unique position, suitable value, and flag. A flag is used to categorise cats in search or trace mode. This work's localizations include M dimensions in the search space with varyinhg speeds. The fitness values represent solution set's (cat's) quality. As a result, before running the algorithm, we must first decide how many cats will be in the loop. The most successful cat in each iteration will be remembered, and the cat from the most recent iteration will act as a success symbol.

3.3.1. Seeking Modes: Resting and Observing

The resting ability of the cat is described in the research mode. In search mode, the cat wanders about the search space while being vigilant. This might be seen as a look for local answers. This mode use the following symbols.

- Seeking Memory Pool (SMP): This parameter specifies how many copies of the chat should be made.
- Seeking Range of selected Dimension (SRD): This stands for differences between new and old sizes of cats selected for mutations.
- Counts of Dimension to Change (CDC): Dimension counts to change in which cats' locations have changed

The steps of seeking mode of CSO algorithm are given as follows.

1. Define counts of copies $(T)\ of\ i^{th}$ cats
2. Do the following based on CDC

a. Randomly add or subtract SRD values from current positions of cats

b. Replace old values for all copies

3. Calculate fitnesses of copies
4. Saelect best solutions and deploy them at positions of i^{th} cats.

3.5.2 Tracing modes

This mode represents hunting capabilities of cats. When cat hunt, positions and speeds of cats are updated and significant disparity between cat's new and previous postures exist. The positions (P_i^c) and Velocities (V_i^c) i^{th} cats in C-dimensional spaces can be defined as $P_i^c = \{P_i^1, P_i^2, \dots . P_i^C\}$; $V_i^c = \{V_i^1, V_i^2, \dots . V_i^C\}$

Where best positions of cats are represented using $P_{best}^c = P_{best}^1, P_{best}^2, \dots . P_{best}^C$

The velocities and positions of jth cats are calculated using (8) and (9).

$$V_{i\,new}^C = w * V_i^c + a * r * (P_{best}^c - P_i^c) \tag{8}$$

where, $V_{i\,new}^C$ represents updated velocities of i^{th} cat in c^{th} dimensions, w represents weight factors in ranges of 0 and 1, V_i^c stands for old velocities of j^{th} cats, a implies user defined constants, r stands for random numbers in ranges of 0 and 1, P_{best}^c sugnifies best achieved position of jth cats in dth dimensions, and P_i^c represents current positions of i^{th} cats in c^{th} dimensions where $c = 1, 2, \dots C$.

$$P_{i\,new}^C = P_i^c + V_i^c \tag{9}$$

where, $P_{i\,new}^C$ signifies updated positions of i^{th} cats in c^{th} dimensions, P_i^cstands for current positions of i^{th} cats in c^{th} dimensions and V_i^c stand for velocities of i^{th} cats.

The CSO algorithm's search and retrieval modes are combined using mixing ratio (MR) which is meant to count cats counts in search and retrieval modes. CSO algorithm's stages are as follows.

1. Initialize the population of cats.
2. Configure user-defined parameters and the number of conversations in search and monitor modes based on the MR parameter value.
3. Determine the ideal posture for each cat's exercise function.

4. According to the flag:

- If the conversation is in search mode, use the search mode procedure;
- otherwise, use the trace mode procedure.

5. According to the MR settings, reset the number of conversations in monitor and search mode.

6. Repetition steps 3–5 till terminating conditions are satisfied.

Problems found in CSO algorithm

The notion of inertial weights is developed as CSO algorithm has premature convergences due to poor varieties [18]. Furthermore, it is reported that the CSO algorithm updates the position of the cats based on their current position and speed. Due of a lack of knowledge on the cat's optimum overall posture, the algorithm may fail to discover an optimal solution. Hence, the following changes to the CSO algorithm are recommended to address these issues.

The ACSO algorithm

- The cat's best overall position is utilised to steer the cat's positions in tracking mode to seek more promising solutions and enhance convergence speeds. Hence, new modified search equations are provided for CSO algorithm's tracking modes, which incorporate optimal overall locations of cats.

$$P_{i\,new}^{C+1} = (1-\alpha) * P_i^c + \alpha * X_g + V_i^c \quad (10)$$

- In tracking mode, the CSO algorithm uses the cat's prior location and velocity vector to update the cat's position. The velocity vector has no effect on the cat's updated location. As a result, to increase diversities of CSO algorithms, particularly in tracking modes, novel update rate equations inspired by

$$V_{i\,new}^{C+1} = V_i^c + \alpha\,(X_g - P_i^c) + \beta * \delta \quad (11)$$

where, δ are uniformly distributed randomized having values in ranges [0, 1]; α and β signify acceleration parameters used to direct positions of cats towards local and global best positions and X_g signify global best positions of cats

To balance explorations and exploitations, adaptibe acceleration parameters α and β are used and computed using following equations.

$$\beta(T) = \beta_{max} - \left\{\frac{\beta_{max}-\beta_{min}}{T_{max}}\right\} * T \quad (12)$$

In (5), β_{max} and β_{min} represent upper and lower limits, T_{max} implies maximum iterations and T stands for current iteration numbers. Hence, $\beta(T)$ are step functions whose values range between upper and lower limits. Larger values of α support explorations while smalles values support exploitations. The aim of parameters $\beta(T)$is to control explorations of cats in search spaces.

$$\alpha(T) = \alpha_{min} + (\alpha_{max} - \alpha_{min}) sin\left\{\frac{\rho T}{T_{max}}\right\} \quad (13)$$

In (13), α_{min} and α_{max} stand for minimum and maximum values of first and last iterations respectively. T_{max} represents maximum iterations and t represents current iteration numbers. The parameter (T) is included to impact the overall exploration capabilities of the suggested method. High values of the parameter $\alpha(T)$ tends to reinforce cat's optimum overall postures while enhancing outcomes.

In pure CSO, a condition must be added to the speed equation to regulate the cat's speed for each dimension and determine if it is within the maximum range. To address this issue, a parameter such as inertial weight will be employed to adjust this section. In this case, the inertial weight value (w) will be determined at random, and experimental findings reveal that w in the range [0.4,0.9] is preferable. Hence, selecting the largest values for w in first iterations (w = 0.9) and reducing them to 0.4 in subsequent iterations. Inertia weighed CSO can converge under certain conditions even without using vmax.

For $w > 1$, values of velocity increase over time, causing cats to diverge beyond the boundaries of search spaces. For w<1, values of velocity decreases over a period of time till it is 0, resulting in convergence behaviors making new global best positional update equations written as

$$V_{K,c} = WV_{k,c} + r_1 d_1 (P_{best} - P_{k,c}) \quad (14)$$

Where d_1 is acceleration coefficient and usually is equal to 2.05 and r_1 is a random value uniformly generated in the range of $[0, 1]$ and w is inertia weight (ACSO).

Positional update equations uses two terms where the first term averages information of current and previous positions while the second averages current and previous velocities (ACSO). The changed global best positional equation can be describes as:

$$P_{i+1} = \frac{P_{i+1}+P_i}{2} + \frac{V_{i+1}+V_i}{2} \quad (15)$$

In this study, ICSO is utilised to increase performance and obtain better convergence with fewer iterations. By adding new parameters to positional equation, such as inertial weights, new velocity equations are created to increase search capabilities for finding locations near best cats. This option allows you to strike a balance between global and local search capabilities. Large inertial mass makes global search easier, whereas little inertial mass makes local search easier. The highest value will be utilised first, then it will progressively decline to the lowest value. As a result, the maximum inertial weight occurs in the first dimension of each iteration and is updated decreasingly in each dimension, allowing the equation that updates the velocity for each cat to the new form to be adjusted. Furthermore, from an optimisation standpoint, the suggested appropriateness calculation technique encourages exploitation and exploration during the search process.

Algorithm 1: ACSO

Input: Medical plant images

Output: Optimal leaf features

Step 1: Initialize parameters of the proposed algorithm including cat counts (n), SMP, SRD, neighborhood structures, $\beta, \alpha \ and \ A$ and randomly placed n cats in random search spaces.

Step 2: Initialise each cat's position and velocity in the C-dimensional search space.

Step 3: Determine the cat's fitness function and memorise the cat's optimal posture.

Step 4: While (i < iterations maximum)

Step 5: Randomly allocate chats into monitor and search modes based on the Flag value.

Step 6: If (Flag==1); Cat in seeking mode.

Step 7: For cats, apply seeking mode processes

Calculate fitness functions for new cat positions.

To control excessive cat movement beyond the search window, choose the average inertial weight (w) at random from the range [0.4, 0.9] each time.

Using (14) calculate a new global best position update.

Using (15), determine the current best place in the world.

Compare the adaptive function's value and recall the cat's optimal posture.

End for

Step 8: Else, Cat in tracing mode

Step 9: For each cat, apply tracing mode process

Update velocities of cats using (10).

Update positions of cats using (11).

Compute fitnesses for newly generated positions of cats

Compare fitness function values and maintain best positions of cats in memory

End for

Step 10: Update positions of cats whiloe determining their best positions

Step 11: Update positions of cats and their best global positions.

Step 12: End if

Step 13: i = i + +

Step 14: End while

Step 15: Obtain final solutions.

The novelty of the ACSO algorithm is increasing the local and global best solutions using average inertia weight values. Hence it is used to classify the accurate results in medical plant dataset

3.3. Medical plant identification and classification using Enhanced Convolutional Neural Network (ECNN) algorithm

An enhanced convolutional neural network (ECNN) is used in this study to identify and categorise the outcomes of the medical tree depending on the chosen characteristics. The ECNN training and testing phases will use the chosen features as input. Low- to high-level features from inputs are extracted by convolutions and downsampling. CNN is one of the most effective deep networks. Essentially, this kind of network has three layers:

fully linked layer, downsampling or pooling layer, and convolution layer. Network access is an optional feature. As illustrated in Figure 3, the network consists of output layers from which systems get trained outputs, input layers that accept features as inputs with intermediate layers being hidden. The suggested ECNN's feature weights are optimised to deliver accurate results. In this study, an enhanced convolutional neural network (ECNN) is utilised to recognise and categorise medical tree findings based on specified characteristics. The selected characteristics will be used as input in the ECNN training and testing phases. CNN, one of the most powerful deep networks, may be composed of numerous hidden layers that recover low-resolution information via convolution and downsampling. This sort of network is made up of three layers:

Convolution layer

In this layer, a kernel (filter) of size aa is convolved with the input image of size RC. The output of the kernel is one pixel, and each block of the input matrix is individually integrated. N output features are created using the convolutional result of the input data and kernel [19]. In general, the filter is the convolution matrix's kernel, and feature maps of size ii are used to describe the characteristics of the output data produced by the convolution kernel and the input data.

The design consists of two convolutional layers, three fully linked layers, and one softmax classification layer. Each convolution layer uses a predetermined number of kernels of different complexity to perform a straightforward convolution operation on the input images. Two convolutional layers, three fully linked layers, and one softmax classification layer make up the architecture. Each convolution layer uses a set number of kernels of varying sizes to conduct a simple convolution operation on the input images. The texture and colour characteristics in the input image are extracted by the seed convolutional layers. These layers generate feature maps from the images fed to them.

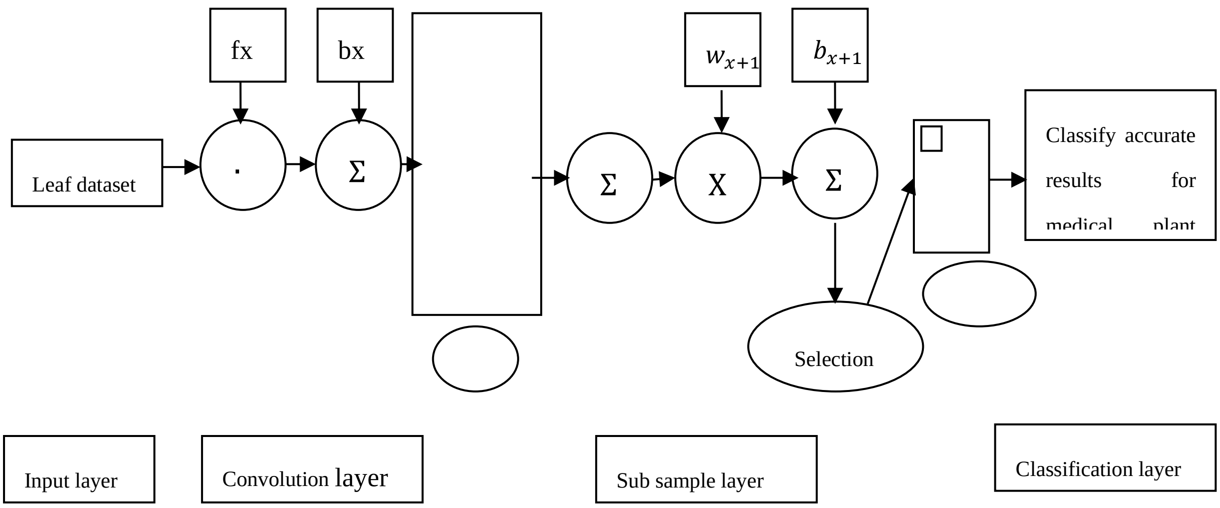

Fig 3 Architecture Diagram of ECNN

ECNN can be composed of numerous convolutional layers, the input and output of which are feature vectors. Each convolutional layer has a set of n filters. These filters are convolved with the input, and the number of filters employed in the convolution process is equal to the depth (n) of the feature map that results.

The quantity and sizes of filters employed in convolution layers affect the sizes of feature map. The first fully connected layers receive outputs of last convolution layers. Following outputs from fully connected layers, the softmax layers classify data and determine probability values of species.

Outputs of l-th convolution layers, denoted by $C_j^{(l)}$ encompass feature maps and are calculated using:

$$C_i^{(l)} = B_i^{(l)} + \sum_{j=1}^{a_i^{(l-1)}} K_{i,j}^{(l-1)} * C_j^{(l)} \quad (16)$$

Where, $B_i^{(l)}$ represents bias matrices and $K_{i,j}^{(l-1)}$ represents convolution filters or kernels of sizes a∗a that connect j-th feature maps with i-th feature maps in same layers. The output $C_i^{(l)}$ layers consist of feature maps. First convolutional layers $C_i^{(l-1)}$ are input spaces, or $C_i^{(0)} = X_i$

Kernels generate feature maps. After convolutions, activations are applied for nonlinear transformations of outputs of convolutional layers:

$$Y_i^{(l)} = Y(C_i^{(l)}) \quad (17)$$

Where, $Y_i^{(l)}$ represents outputs of activation functions (Sigmoids, tanh, and rectified linear units) and $C_i^{(l)}$ represents received inputs. In this study, ReLU is indicated as Y_i((l))=max(0,Y_i((l))). This feature is frequently employed in DL models since it aids in the reduction of interaction and nonlinear effects. If the input is negative, ReLU changes the output to 0; otherwise, input values are returned. The benefit of this activation function over others is that it learns quicker since error derivatives become extremely small in the saturation area, resulting in essentially no adjustments of weights. This is referred to as the vanishing gradient problem.

Sub sampling Layer

This layer's primary purpose is to reduce spatial dimensions of feature maps derived from preceding convolutional layers where masks of size b x b are selected, and downsampling operations are conducted between masks and feature maps. It is important to note that downsampling layers assist convolution layers in tolerating rotations and translations across input images. The ideal weights are changed in this suggested study activity based on the average of the feature weights.

Weighted mean $w_H = \frac{N}{\sum_{i=1}^{N} wx_i}$ (11)

Where,

N − Number of features

w- Weight value of the feature

x_i- Features

Full Connection

The output layer uses Softmax activation function:

$$Y_i^{(l)} = f(z_i^{(l)}), \text{ where } z_i^{(l)} = \sum_{i=1}^{m_i^{(l-1)}} w_H y_i^{(l-1)} \quad (18)$$

where $w_H$are weighted Harmonic means of features that should be tuned by fully connected layers for representations of classes and f stands for transfer functions representing non-linearities. Finally, classifiers connect fully connected layers and output layers to complete t medical image classification selections.

Steps of ECNN

1. Dataset of medicinal plant leaves
2. Describe the functioning of the medical table given data set for all input attributes.
3. Subclassify the input
4. Recognise leaf features
5. Using the characteristics, extract more interesting and relevant features.
6. Carry out the training and testing steps for the provided leaf image dataset.
7. For each feature in the input dataset, copy the preset leaf feature labels.
8. Improve the accuracy with which results on medicinal plants are identified and classified

Experimental result

Santalum album (sandalwood), Muntingia calabura (Allspice cherry), Plectranthus amboinicus / Coleus amboinicus (Indian mint, Mexican mint), Brassica juncea (oriental mustard), and many others are included in the dataset. The collection contains 1,500 photos of forty different species. Each species has 60 to 100 high-quality images. The files are called after the species' botanical/scientific name. For the Matlab experiment, we photographed neem, mint, tulsi, sandalwood, and fenugreek trees. The performance criteria under consideration, including as accuracy, recall, precision, and f-measure, are assessed using current CNN [20] and suggested ACSO+ECNN algorithms..

Accuracy

Accuracies are determined as overall correctness of models and are computed as total actual classification parameters $(T_p + T_n)$ which are segregated by sums of classification parameters $(T_p + T_n + F_p + F_n)$. The accuracy is computed as like :

$$\text{Accuracy} = \frac{T_p+T_n}{(T_p+T_n+F_p+F_n)} \quad (19)$$

Where Tp is true positive, Tn is true negative, Fp is false positive and Fn is false negative

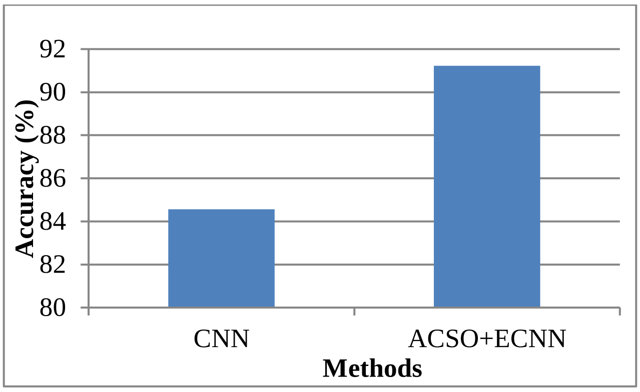

Fig 5 Accuracy

The comparative measurement accuracy is evaluated using the existing approach and the suggested method, as shown in Figure 5. The x-axis is used to depict data sets and processes, whereas the y-axis is used to plot exact numbers. Existing approaches, such as the CNN algorithm, give lesser accuracy for various medical tree identification and classification datasets, however the suggested ACSO+ECNN methodology provides superior accuracy. The preprocessing strategy enhances classification accuracy by addressing the noise reduction issue. The findings suggest that by choosing the optimal features, the proposed ACSO+ECNN algorithm enhances the classification accuracy of the medicinal plant dataset.

Precision

The precision is calculated as follows:

$$\text{Precision} = \frac{T_p}{T_p+F_p} \quad (20)$$

While recalls are measurements of comprehensiveness or quantities, accuracies can be conceived as computations of precisions or quality. High precisions often mean algorithms provide more relevant results than irrelevant ones. The accuracies of classified classes are computed by dividing the total labelled object counts that fall into positive classes by counts of genuine positives.

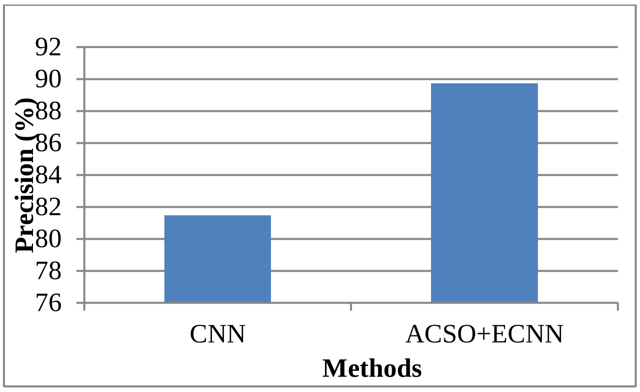

Fig 6 Precision

As illustrated in Figure 6, the existing approach and the suggested method are used to compare measurement accuracy. On the x-axis, the techniques are employed, and the precise result is given on the y-axis. The suggested ACSO+ECNN approach outperforms the traditional CNN algorithm, which is less accurate, on several medical worksheet datasets. The suggested method improves accuracy by picking more appropriate data. Because of its excellent properties, the suggested ACSO+ECNN algorithm increases the classification accuracy of medicinal plant leaves..

Recall

The calculation of the recall value is done as follows:

$$\text{Recall} = \frac{T_p}{T_p+F_n}$$

(21)

The comparison graph is depicted as follows:

As opposed to precision values defined as counts of relevant samples obtained by searches divided by total retrieved document counts in searches, recalls are defined as counts of relevant documents retrieved in model searches.

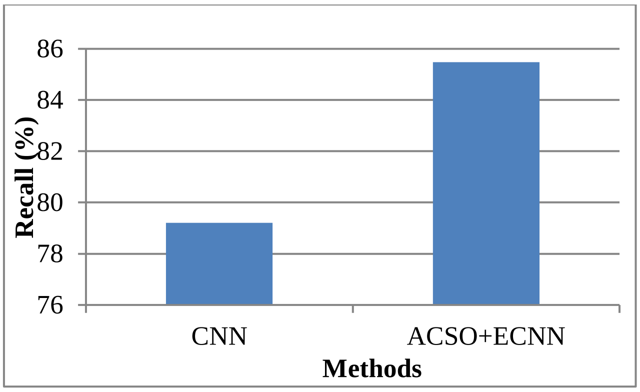

Fig 7 Recall

As noted in Figure 7 above, both old and new approaches are assessed in terms of recall for the comparison metrics. The approaches form the x-axis, and recall values the y-axis. The suggested ACSO+ECNN method offers greater recall compared to the existing CNN algorithms for provided dataset of medical leaves which have lesser recall. It improves training steadiness and performance. As a result, the dataset's training process is substantially more stable.

F-measure

F-measures are combinations of values of precisions P and recalls R,

$$F = 2.\frac{PR}{P+R} \tag{22}$$

Assessments of classifications depend on values of F-measure as they are standard measures of summarizing precisions P and recalls R

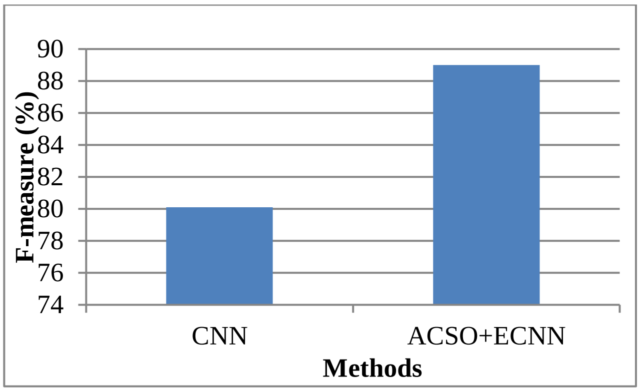

Fig 8 F-measure

The comparison values for the F-measure utilising the existing and suggested techniques will be examined using Figure 8. For the supplied medical leaf image dataset, the present CNN delivers a lower F-measure, but the suggested ACSO+ECNN method provides a higher F-measure. The suggested classifier predicted with an F1 score of 89% and no misinterpreted features. To provide optimal functioning, the ACSO algorithm is employed. As a result, the suggested technique improves classification accuracy and performance for the provided image dataset.

4. Conclusion

The ACSO+ECNN method is suggested in this study to enhance the performance of the supplied data set's medicinal plant categorization. Preprocessing, feature extraction, feature selection, and classification are the four primary elements in this study. By managing noisy values, preprocessing seeks to raise the quality of the data collection. In order to extract the most informative features, feature extraction was carried out utilising QH and GLCM. The ACSO algorithm is used to choose features, and it efficiently chooses the best features. The ACSO algorithm therefore offers practical and appropriate capabilities for usage in real-time applications. Finally, the ECNN algorithm-based categorization offers a performance that is more accurate. The suggested ACSO+ECNN model enhances medical diagnostics. Based on the results of the experiments, it can be said that the suggested ACSO+ECNN algorithm is less accurate, precise, recallable, and time-consuming than existing methods. Edge and shape aspects can be taken into consideration for future work to improve performance.

References

[1] Altemimi, Ammar, et al. "Phytochemicals: Extraction, isolation, and identification of bioactive compounds from plant extracts." *Plants* 6.4 (2017): 42.

[2] Naeem, Samreen, et al. "The classification of medicinal plant leaves based on multispectral and texture feature using machine learning approach." *Agronomy* 11.2 (2021): 263.

[3] Nabavi, Seyed Fazel, et al. "Antibacterial effects of cinnamon: From farm to food, cosmetic and pharmaceutical industries." *Nutrients* 7.9 (2015): 7729-7748.

[4] Lee, Sue Han, et al. "How deep learning extracts and learns leaf features for plant classification." *Pattern Recognition* 71 (2017): 1-13.

[5] Nelly, Arnold, Delelis-Dusollier Annick, and Dupont Frederic. "Plants used as remedies antirheumatic and antineuralgic in the traditional medicine of Lebanon." *Journal of ethnopharmacology* 120.3 (2008): 315-334.

[6] Chuanlei, Zhang, et al. "Apple leaf disease identification using genetic algorithm and correlation based feature selection method." *International Journal of Agricultural and Biological Engineering* 10.2 (2017): 74-83.

[7] Kumar, Arun, et al. "Optimizing feature selection using particle swarm optimization and utilizing ventral sides of leaves for plant leaf classification." *Procedia Computer Science* 89 (2016): 324-332.

[8] Kan, H. X., L. Jin, and F. L. Zhou. "Classification of medicinal plant leaf image based on multi-feature extraction." *Pattern Recognition and Image Analysis* 27 (2017): 581-587.

[9] De Luna, Robert G., et al. "Identification of philippine herbal medicine plant leaf using artificial neural network." *2017IEEE 9th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment and Management (HNICEM)*. IEEE, 2017.

[10] Anton, Anton, et al. "Classification of Tomato Plant Diseases Through Leaf Using Gray-Level Co-occurrence Matrix and Color Moment with Convolutional Neural Network Methods." *Smart Trends in Computing and Communications: Proceedings of SmartCom 2020*. Springer Singapore, 2021.

[11] Gopal, A., S. Prudhveeswar Reddy, and V. Gayatri. "Classification of selected medicinal plants leaf using image processing." *2012 International Conference on Machine Vision and Image Processing (MVIP)*. IEEE, 2012.

[12] Pankaja, K., and V. Suma. "Plant leaf recognition and classification based on the whale optimization algorithm (WOA) and random forest (RF)." *Journal of The Institution of Engineers (India): Series B* 101 (2020): 597-607.

[13] Quoc, Trung Nguyen, and Vinh Truong Hoang. "Medicinal Plant identification in the wild by using CNN." *2020 International Conference on Information and Communication Technology Convergence (ICTC)*. IEEE, 2020.

[14] Verma, Kesari, Bikesh Kumar Singh, and A. S. Thoke. "An enhancement in adaptive median filter for edge preservation." *Procedia Computer Science* 48 (2015): 29-36

[15] Ibrahim, Haidi, Nicholas Sia Pik Kong, and Theam Foo Ng. "Simple adaptive median filter for the removal of impulse noise from highly corrupted images." *IEEE Transactions on Consumer Electronics* 54.4 (2008): 1920-1927

[16] Benčo, Miroslav, and Robert Hudec. "Novel method for color textures features extraction based on GLCM." *Radioengineering* 16.4 (2007): 65

[17] Lin, Kuan-Cheng, et al. "Feature selection based on an improved cat swarm optimization algorithm for big data classification." *The Journal of Supercomputing* 72.8 (2016): 3210-3221.

[18] Orouskhani, Meysam, et al. "A novel cat swarm optimization algorithm for unconstrained optimization problems." *International Journal of Information Technology and Computer Science* 5.11 (2013): 32-41

[19] Barbosa, Alexandre, et al. "Modeling yield response to crop management using convolutional neural networks." *Computers and Electronics in Agriculture* 170 (2020): 105197

[20] Dileep, M. R., and P. N. Pournami. "AyurLeaf: a deep learning approach for classification of medicinal plants." *TENCON 2019-2019 IEEE Region 10 Conference (TENCON)*. IEEE, 2019.