Vol.68: e25240881, 2025
https://doi.org/10.1590/1678-4324-2025240881
ISSN 1678-4324 Online Edition

BABT
Brazilian Archives of Biology and Technology

Article - Biological, Medical and Health Technologies

DeepCNNMed: Enhancing Medicinal Plant Identification Through Deep Convolutional and Hybrid Neural Networks

Rajalakshmi Gurusamy[1*]
https://orcid.org/0000-0003-1722-373X

Parameswari Chandrasekaran[2]
https://orcid.org/0000-0001-9433-5765

Yamuna Devi Manickam Malu[3]
https://orcid.org/0000-0002-8258-7882

Madhusudhanan Sampath[4]
https://orcid.org/0000-0003-4134-8137

Nesarani Abraham[5]
https://orcid.org/0000-0002-7604-9957

[1]Sethu Institute of Technology, Department of Information Technology, Tamilnadu, India; [2]Sethu Institute of Technology, Department of Information Technology, Tamilnadu, India; [3]Koneru Lakshmaiah Education Foundation, Department of Computer Science and Engineering, Andhra Pradesh, India; [4]Rajalakshmi Engineering College, Department of Computer Science and Engineering, Tamilnadu, India; [5]Koneru Lakshmaiah Education Foundation, Department of Computer Science and Engineering, Andhra Pradesh, India

Editor-in-Chief: Paulo Vitor Farago
Associate Editor: Jane Manfron Budel

Received: 17-Oct-2024; Accepted: 13-Mar-2025

*Correspondence: gurusamyrajalakshmi89@gmail.com (R.G.).

HIGHLIGHTS

- First, the original image was converted into a threshold image and segmented veins.
- Texture and color features are extracted from thresholded and segmented vein images using GLCM.
- DeepCNN was trained using the texture and color features returned from GLCM using RPNN+FCNN techniques.
- The accuracy of DeepCNN model is 97.2% with negligible real-time image loss.

Abstract: There are innumerable types of plants, many of which have therapeutic applications. Traditional medicine frequently makes use of medicinal herbs. Accurate identification of medicinal plants would be highly advantageous to the forest service, life scientists, physicians, pharmaceutical companies, governments, the public, and the people who take drugs. Manual methods are quick and accurate for identifying plants, but only subject-matter specialists should use them. However, it does take some time. False positives are possible, and they could not only cause serious problems but also negative consequences. Deep learning techniques are rapidly being used to solve computer vision challenges. In this study, we suggested DeepCNNMed (RPNN + FCNN), an automated system for categorizing medicinal plants that will enable individuals to recognize valuable plant species fast. It is well known that feature extraction and classification are impacted by RPNN + FCNN. The RPNN + FCNN classifier was used to separate the exception properties and classify them, and the resulting DeepCNNMed model exhibits 97.2% accuracy with negligible losses on real-time images.

Keywords: Radial Basis Probabilistic Neural Network (RPNN); Fully Connected Neural Network (FCNN); Grey Level Co-occurrence Matrix (GLCM); Computer Vision; Medicine Plant Identification.

INTRODUCTION

Plants are multicellular organisms on the planet that feed them through photosynthesis. There are about 300,000 different kinds of plants, with common examples include plants such as trees, shrubs, and grasses. A significant part of the ecosystem on Earth is played by plants. Numerous species are crucial to the food chain and also contribute significantly to the production of oxygen on Earth since they consume plants and other herbivorous animals. Indian medicinal plants are typically identified using arduous, time-consuming, and challenging traditional methods that are frequently wrong.

The similarities between and within plant classes, the potential for complex backgrounds, and variations in many traits, including lighting and colour, make it difficult to classify plants using digital pictures of leaves. Therefore, it is important to create methods and tools that can effectively analyses and interpret patterns in leaf pictures. Much recent research has not focused on developing models and methods to speed up the automatic recognition of Indian herbal medicines. The public and others interested in growing medicinal plants can make their knowledge easier through the computerized identification of medicinal herbs, instead of relying on professional botanists and experts in Ayurveda (the ancient Indian medical system).

Since plant leaves are only two-dimensional, image processing techniques can be used to automatically identify plant leaves. Plant classification is one computer vision problem that has recently proven particularly effective at solving using deep learning (DL), or CNNs. DL eliminates the need for basic feature extraction and subject understanding that only expert botanists can provide.

This study's major goal is to quickly forecast outcomes while accurately classifying and identifying photos of luxuriant medicinal plants. Deep CNN models in plant identification systems are used to identify and classify different plant species based on specific leaf images. A custom medicinal leaf dataset consisting of 500 images identifying 10 species of Indian herbs (Ayurveda) is used to form and pre-process the leaf images. Using image pre-processing techniques, the image is initially converted into a suitable format that simplifies feature extraction. Then, extract features that are used to classify plants using several CNN layers.

The organizational structure of the research is as follows. Previous studies on plant taxonomy are described in Section II. A recommended DeepCNNMed structure is provided in Section III. Results and description of the proposed method are presented in Section IV. The conclusions of the survey are presented in Section V.

LITERATURE REVIEW

Modern computer vision techniques that can be used to identify medicinal plants are theoretically outlined in this section. We start by outlining traditional leaf detection techniques used in picture processing. Based on morphological and textural traits, a feature extraction technique was proposed by Mahajan, S. and coauthors [1] for plant recognition. Applying the proposed approach to a self-generated dataset of 20 plant species results in good accuracy. This method, however, is unable to distinguish between leaves with similar shapes, leaves that have been shaded, and faulty leaves.

Zhang Y and coauthors [2] employed a characteristic bag (BOF) to achieve an overall peak accuracy of 94.22%. A controlled and thoroughly examined comparison assessment of numerous research and models using different bases, such as CNN and mathematical based learning, etc. was published by Azlah and coauthors in [3]. They discussed a number of plant leaf classification techniques, including support vector machine (SVM), CNN, and kNN, providing key insights for developing strategies to achieve this goal. The authors discussed the properties of classifiers and emphasized that CNNs are not suitable for intensive processing and lack generalization.

Another important study was carried out by Kumar and coauthors [4] used various classification models including kNN, decision trees, and multi-layer perceptron along with the AdaBoost algorithm and achieved an accuracy rate of 95.42%. Deep convolution network-based classifier Google Net was developed by Jeon, W.S and coauthors [5], who also provided helpful ideas on how to employ CNNs for this purpose. Their system had a recognition rate of more than 94%. Pankaja and coauthors [6] used a PCA classifier to recover features based on texture and shape, achieving an achievable accuracy of 96.66%.

The multilayer perceptron (MLP) decoder [7] trained over morphologically produced an efficiency of 94.0%. Kadir, A and coauthors [8] attained an accuracy of 93.75% using the same morphological features. Priya and coauthors [9] achieved an accuracy of 94.20% with a SVM classification model using morphology, vein structure, and geometric information. Anami [10] and Sambhaji [11] achieved accuracies of 93.6% and

85%, respectively, using leaf recognition. Sun and coauthors [12] achieved a recognition rate of 91.78% on the BJFU100 dataset using DL model.

In recent years, DL approaches are widely used because they can handle both feature extraction and image classification. DL has become increasingly popular in various automation applications that require image segmentation and classification [17–20]. One of the most effective deep learning techniques is the CNN, which is known for its extraordinary effectiveness in pattern recognition and image segmentation. CNNs use carefully trained layers that have been used in several studies to identify and classify plants [21] and identify plant diseases [22].

CNNs consist of a hierarchy of self-learning properties. Deep layers learn high-level features such as textures and objects in the image, while fast layers learn low-level features such as color, corners, and edges. These models combine two key stages of the image processing process: feature extraction, learning, and classification. Thus, unlike traditional machine learning techniques, manual feature extraction is not required. Using digitized images and three DL models (CNN, VGG19, and VGG16), Dey and coauthors [23] were able to identify 64 different medicinal plants with 95.7%, 97.6%, and 97.8% accuracy, respectively. Pushpa and Rani [24] performed a comparative study of Resnet34, Densenet121, MobileNetV3Large, VGG16, and Resnet50 for Ayurvedic plant identification using the deep convolutional neural network Ayur-PlantNet. In particular, Ayur-PlantNet achieved the highest accuracy of 92.27%. A comparison of the performance of different models is shown in Table 1.

Previous studies [25, 26] mainly relied on public datasets and there is a research gap on identifying medicinal plants from raw data with complex contexts from various geographical distributions. Variations in leaf shape and dimensions can have a significant impact on accurate identification in real-world conditions. Although some studies have used real field datasets with complex conditions [27], extensive and rigorous research is needed to test and build models for rapid and accurate identification of medicinal plants in diverse geographical field circumstances. Furthermore, even with vision-based systems that has greatly enhanced their capacity to extract complicated traits and choose the most meaningful ones using sophisticated DL algorithms, distinguishing therapeutic herbs from other plants remains a challenge.

This research gap needs to be filled to better understand the strengths and weaknesses of different medicinal plant identification algorithms. Since the DeepCNN algorithm (RPNN + FCNN) was developed and trained using the texture and color features returned by GLCM, the main objective of the current work was to establish a real-time automatic system for medicinal plant recognition. The proposed DeepCNN model achieved an accuracy of 97.2% with little frame loss in real time.

Table 1. Performance comparison of plant leaf identification based on some recent studies

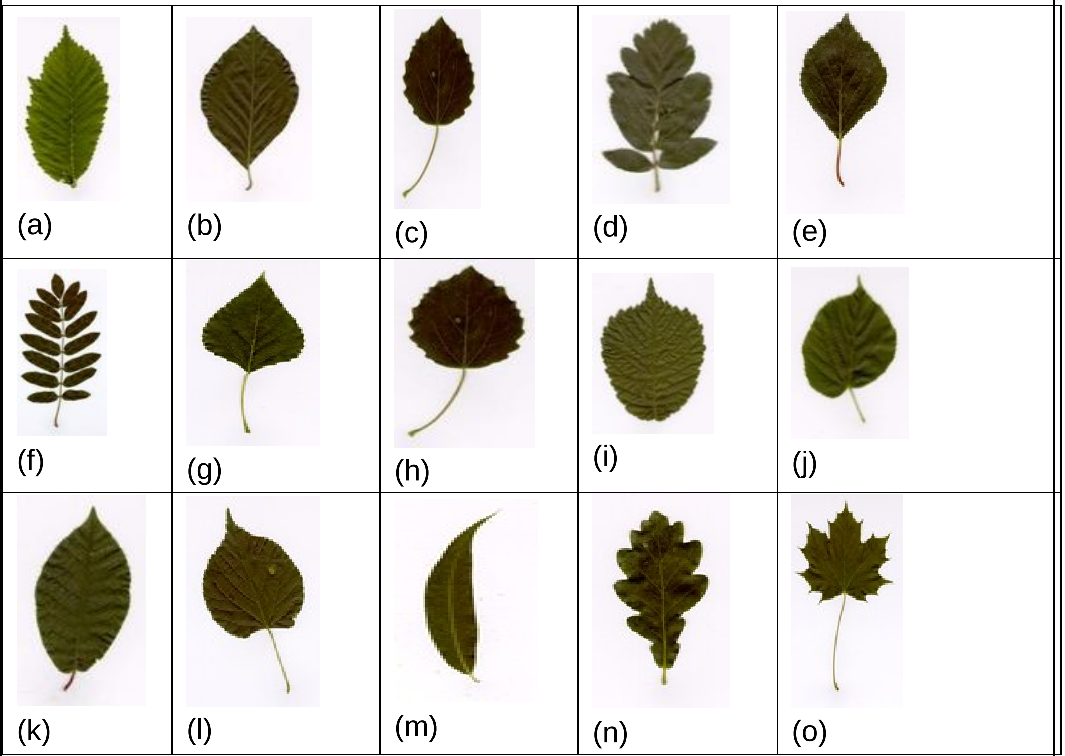
Dataset	Feature Selection Method	Model	Accuracy (%)
30 medicinal plant species across 20 families. [23]	Automatic feature extraction -CNN	CNN	95.7
		VGG16	97.8
		VGG19	97.6
GSL100 leaf dataset. [24]	Fusion feature model	Hierarchical CNN	94.54
RTP40 dataset. [24]			75.46
SaudiArabiaFlora Dataset. [25]	SHAP (SHapley Additive exPlanations)	MIV-PlantNet (MobileNet+ Inception+ VGG)	96
Indian Ayurvedic plant species. [28]	Automatic feature extraction -CNN	DCNN	92.27
Private Dataset. [29]	Automatic feature extraction -CNN	DenseNet	97.3
Plant Village (Tomato) [30]	Multi-Level Feature Fusion	InceptionV3	94.5
		MobileNetV1	82.7
		MobileNetV2	92.1
		Modified-Xception	99.61
Private Dataset. [31]	Kernel method	CNN	90

MATERIAL AND METHODS

The proposed DeepCNNMed performs the subsequent actions to correctly categories the identification of the medicinal plant.

IMAGE ACQUISITION

Table 2. Tree classes of 15 species

Tree classes	Single Image
a) *Salix aurita* L	
b) *Fagus sylvatica* L	
c) *Ulmus carpinifolia* Gled	
d) *Sorbus intermedia* A.Blytt	
e) *Alnus incana* (L.) Moench	
f) *Sorbus aucuparia* L	
g) *Tilia* L	
h) *Populus tremula* L	
i) *Ulmus glabra* Huds	
j) *Betula pubescens* Ehrh	
k) *Salix* L	
l) *Populus* L	
m) *Salix alba* L	
n) *Quercus* L	
o) *Acer* L	

Plant identification involves taking an image of the plant, which is made easier by scanning or pseudo-scanning as the background is captured smoothly. Standard datasets such as Swedish (15 leaf species), Flavia (32 leaf species), and ICL (220 plant species) are readily available. Images from these three datasets have been used in the majority of studies. In our investigation, we used a Swedish dataset of 75 images for each of the 15 plant species for a total of 1,125 images. The names of the 15 species are included in Table 2, along with a single image for each species.

LEAF IMAGE PRE-PROCESSING

Image pre-processing is crucial to improve image quality for post-processing, as inherent noise in images reduces classification accuracy. It is designed to handle corrupted data and remove noise that interferes with identification. In this study, RGB conversion, binary conversion, smoothing, filtering, noise management, scaling, and image enhancement are used to enhance the leaf images using pre-processing methods including noise management. The overall structure of the pre-processing techniques is shown in Figure 1.

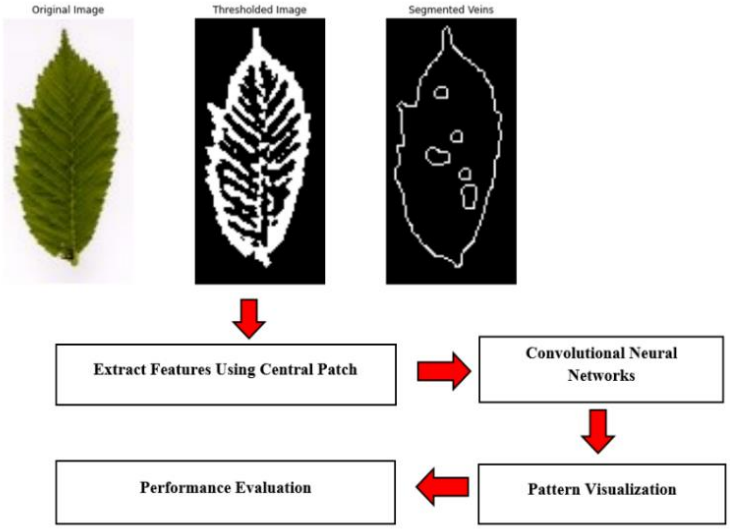


Figure 1. DeepCNNMed Leaf Classification Architecture

Gaussian Filtering

In this work, Gaussian filtering (also known as Gaussian smoothing) was used to process noise in the leaf images. This is a linear filter that removes superfluous or noisy information from an image. Equation 1 contains an expression that applies a Gaussian filter.

$$G_{Smooth\_images} = \frac{1}{2\pi\sigma^2} e^{(h^2+v^2)\frac{1}{2h^2}} \tag{1}$$

where, σ = Standard deviation, h = horizontal axes, v = vertical axes.

Scaling

After noise reduction processing, resizing was performed. The image scale was set to [300 × 400], as shown in Equation 2.

$$Resized(G) = Resize(G_{Smooth\_images}, [300,400]) \tag{2}$$

Image Enhancement

When working with colour images, the image processing process becomes important. In addition, the next phase consists of segmentation of colour images, and the image contrast and texture should be improved for better results. Before segmentation, image enhancement removes all extra pixels from colour images.

VEIN SEGMENTATION

Regional Proposal Neural Network (RPNN)

The RPNN module uses a sliding window to select from over 200,000 anchor points at the feature map locations and resizes the frame to fit the object. By keeping the highest scoring anchor points in the foreground and removing duplicate anchor points, the accuracy and efficiency of image detection is improved. It also reduces false positives by filtering out unnecessary anchor points.

ROI Align

By utilising bilinear interpolation, this can crop a section of the feature map that corresponds to the Region of Interest (ROI) and send that data to the ROI classifier and bounding box regression.

Box Regression and Classification

The ROI refinement phase includes an ROI classifier and bounding box regression to improve classification accuracy by extracting complex features. These methods improve ROI localization, enabling more accurate object detection. The combination of ROI classification and bounding box regression ensures accurate identification and localization of objects in an image, facilitating subsequent analysis and decision-making.

Segmentation Mask

The ROI classifier's positive ROI areas can be hidden using this branch's convolutional network. In order to keep light from the mask's branches (or pixels), a lower-resolution mask was produced. However, the mask is resized to match the output image's bounding box dimensions.

FEATURE EXTRACTION

Feature extraction is a crucial step in image processing and pattern analysis because it reduces the number of variables required for post-processing. This involves pre-processing and segmenting the image to identify regions of interest, thereby reducing redundant data and memory usage. The research combined texture and colour features were combined to render the image.

Texture Features

In this study, statistical methods such as 1st order, 2nd order, and mth order methods are used to define the texture of grey scale images. Grey level co-occurrence matrix (GLCM) is used to extract texture features by considering spatial relationships. Five characteristics are used: energy, correlation, entropy, contrast, and uniformity. The formulas for calculating these characteristics are shown in Table 3.

Table 3. Texture features formula

Feature Name	Formula
Energy	$\sum_{l,m} \mathrm{Pro}_{\mathrm{dist}}(l,m)^2$
Correlation	$-\sum_{l,m} \frac{(l-\mu)(m-\mu)}{\sigma^2} \mathrm{Pro}_{\mathrm{dist}}(l,m)$
Entropy	$-\sum_{l,m} \mathrm{Pro}_{\mathrm{dist}}(l,m) \log \mathrm{Pro}_{\mathrm{dist}}(l,m)$
Homogeneity	$\sum_{l,m} \frac{1}{1+(l-m)^2} \mathrm{Pro}_{\mathrm{dist}}(l,m)$
Contrast	$\sum_{l,m} (l-m)^2 \mathrm{Pro}_{\mathrm{dist}}(l.m)$

Colour Features

In this study, the input image is split into three colour channels for segmentation and colour features such as mean, variance, kurtosis, and skewness are extracted. These features are used to classify different regions of the image. The segmentation process helps to identify patterns of interest in the image for further analysis. These features are known as colour-based texture features and their formulas are given in Table 4.

DeepCNNMed LEAF CLASSIFICATION

Previously, high-resolution images were difficult to use with multi-layer perceptron (MLP) models due to the global connections between nodes. DeepCNNs solve this problem by converting the input image into a single feature vector and using hidden layers. This allows DeepCNNs to efficiently learn a spatial hierarchy of image features, making them better suited for high-resolution image analysis tasks. DeepCNN's hierarchical structure organizes neurons in three dimensions, converting 3D volumetric inputs into output activations.

Table 4. Colour features formula

Feature Name	Formula
Mean	$\frac{\sum_{i=1}^{N} x_i}{N}$
Variance	$\frac{\sum_{i=1}^{N} (x_i - \mathrm{mean})^2}{N}$
Kurtosis	$\frac{1}{N}\sum_{i=1}^{N} \{\frac{x_i - \mathrm{mean}}{\mathrm{sqrt(variance)}}\}^3$
Skewness	$\frac{1}{N}\sum_{i=1}^{N} \{\frac{x_i - \mathrm{mean}}{\mathrm{sqrt(variance)}}\}^4$

Algorithm 1: DeepCNNMed (RPNN+FCNN)

Input: Training set X_{train}, y_{train} , and test set X_{test}
Output: Weight & bias of Radial Basis Probabilistic Neural Network (RPNN) and Fully Connected Neural Network (FCNN)

Begin:

Step 1: Pre-process the input data : Normalize X_{train} and X_{test}
Step 2: Train the RPNN (Radial Basis Probabilistic Neural Network)
i) Define RBF centres (C_i) and bandwidth (σ)
ii) For each input x in X_{train}:
Compute φ(x,C_i) for each center C_i as follows:

$$\varphi(x, C_i) = \exp(-\frac{|x-C_i|^2}{2\sigma^2}) \quad (3)$$

iii) Compute class probabilities P (C_j | x) for each class C_j:

$$P(C_j|\, x) = \sum_{i=1}^{n} p(\frac{C_n}{x}) \quad (4)$$

Step 3: Train FCNN

i) Initialize feed forward FCNN with number of RPNN features($P(C_j|\, x)$), activation function (ReLU), and output layer (softmax).
ii) FCNN training using RPNN features as input and y_{train} as labels.
iii) Apply back propagation and optimization (Adam) algorithm.

Step 4: Make Prediction
For each X_{test}, compute RPNN features using RPNN, feed RPNN features to FCNN.

Step 6: Performance Evaluation (accuracy and F1 score).
Step 7: End

This allows DeepCNNs to capture local patterns through convolutional layers and learn global patterns through pooling layers. DeepCNNs are particularly effective for tasks image segmentation because of their ability to extract features at different levels of abstraction.

Parameter Tuning Process of RPNN+FCNN

When combining RPNN with a FCNN, hyperparameter tuning is essential for optimizing model performance. The hyperparameter of the RPNN and FCNN is shown in Table 5.

Table 5. List of parameters

Method	**Parameters**	**Values**
RPNN	Radial Basis Width (σ)	{0.1, 0.5, 1, 2}
	Number of Centers (C_i)	{50, 100, 200}
	Optimization Strategy	Grid Search
FCNN	Number of Layers (L)	{1, 2, 3, 4, 6, 8}
	Number of Neurons per layer (N_l)	{32, 64, 128}
	Activation Function (τ)	{ReLu, Tanh, Sigmoid}
	Learning Rate (η)	{ 1e-5, 1e-1}
	Optimizer (o)	{Adam, SGD, RMSprop}
	Batch Size (B)	{32, 64, 128, 256}
	Dropout (λ)	{0, 0.1, 0.2}
	Optimization Strategy	Grid Search

To improve learning and generalizing discriminative features on challenging datasets, the hybrid model that integrates probabilistic feature extraction of RPNN with the powerful representation learning of FCNN. To achieve model performance on the training dataset, RPNN + FCNN performs hyperparameter tuning by following steps shown in Algorithm 2.

Algorithm 2: RPNN + FCNN Hyperparameter Tuning Process

Step 1: Optimize RPNN parameters such as RPNN $\epsilon\ \sigma$, C_i parameters using grid search technique. It returns the feature space of FCNN.
Step 2: The RPNN feature space is applied to FCNN input. Optimize FCNN parameters such as FCNN ϵ L, N_l, τ, η, o, B, λ by using RPNN feature space. Parallel computing is used to speed up the grid search process.
Step 3: After fine-tuning, to maximize the model accuracy, the optimal hyperparameters are used to train the final model with a Bayesian optimization approach.
Step 4: Finally, we evaluated the performance of different models using various model performance metrics such as Accuracy (Acc), Positive Predictive Value (PPV), True Positive Rate (TPR), and F1-score.

EXPERIMENTAL RESULTS AND DISCUSSION

DATASET

In this study, a hybrid model is trained using the widely-published Swedish leaf dataset. This is because it provides scanned images with white backgrounds with little or no pre-processing required. For our experiments, the Swedish Leaf dataset was used, which contains 1,125 images of leaves from 15 different plant species and can be accessed at https://www.cvl.isy.liu.se/en/research/datasets/swedish-leaf. To achieve uniformity and stability of the model, the images are scaled to a size of 64 x 64 pixels. The dataset is split into 80% training set of 900 images used to train the model and 20% test set of 225 images used to evaluate the model performance.

PERFORMANCE MEASURES

The proposed RPNN + FCNN hybrid method outperforms both the standalone RPNN and FCNN techniques shown in figure 3. The model performance is evaluated using four different metrics: Accuracy (Acc), True Positive Rate (TPR), Positive Predictive Value (PPV) and F1-score, which are defined as follows:

$$\mathrm{Acc} = \frac{\mathrm{TP+TN}}{\mathrm{TP+TN+FP+FN}} \tag{5}$$

$$T\mathrm{PR} = \frac{\mathrm{TP}}{\mathrm{TP+FN}} \tag{6}$$

$$PPV = \frac{TP}{TP+FP} \tag{7}$$

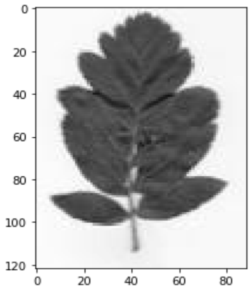
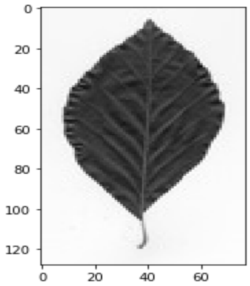
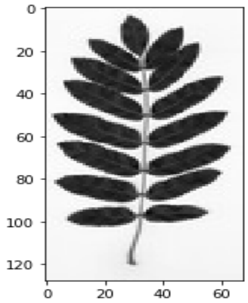
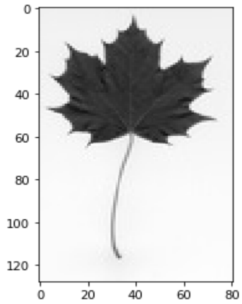
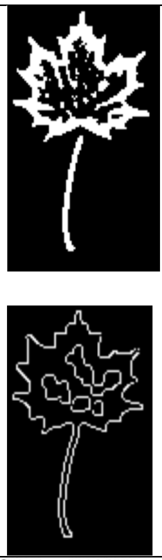
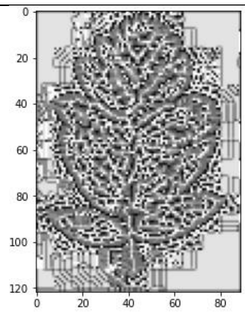
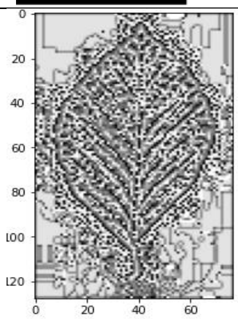
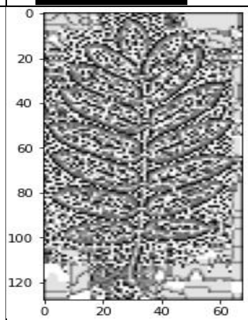
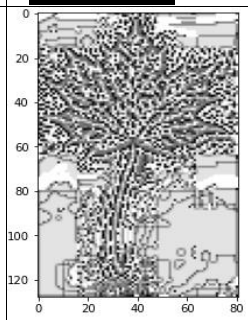
$$F1 = 2 * \frac{PPV*TPR}{PPV+TPR} \tag{8}$$

Where, TP = True Positive, TN = True Negative, FP = False Positive, FN = False Negative

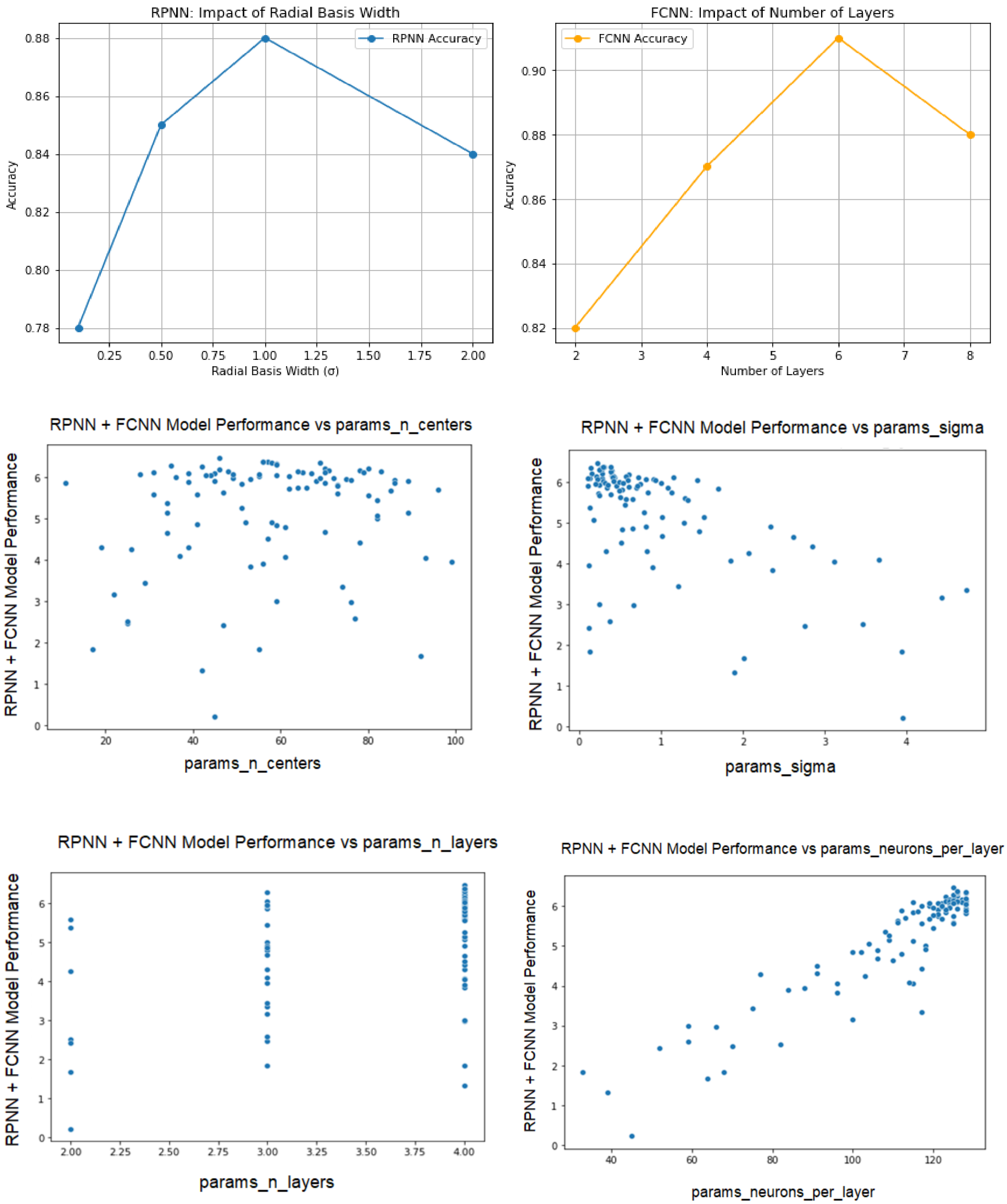
EXPERIMENTAL RESULTS

The DeepCNN classifier, trained with FCNN and RPNN features effectively identifies plant species through three main classification steps: image pre-processing, texture extraction, and colour features recognition shown in Table 6.

Table 6. Leaf image pre-processing: original image, image to grey, threshold image & segmented veins, texture feature visualization, texture and colour features

Input Image				
RGB2Gray				
Threshold Image & Segmented Veins				
Texture Features Visualization				
Texture Feature	Contrast : 2553.3731 Correlation : 0.8173 Energy: 0.0833 Homogeneity: 0.2508	Contrast : 3477.2421 Correlation : 0.8022 Energy: 0.0980 Homogeneity: 0.3155	Contrast: 7529.0489 Correlation: 0.5878 Energy: 0.0632 Homogeneity: 0.2345	Contrast: 2167.7081 Correlation: 0.8359 Energy: 0.1445 Homogeneity: 0.4462
Colour Feature	Mean: 170.2810 Variance: 6963.886 Skewness: -0.2376 Kurtosis: -1.8546	Mean: 175.1634 Variance: 8598.084 Skewness: -0.5112 Kurtosis: -1.6875	Mean: 170.6497 Variance: 8930.129 Skewness: -0.5417 Kurtosis: -1.5885	Mean: 202.7452 Variance: 6320.852 Skewness: -1.3068 Kurtosis: -0.2059

Each convolutional layer filter has its own visual characteristics, such as vertical and horizontal borders, colour, texture, and density. To avoid overfitting, dropout, and L2 regularization techniques are used in the feature extraction process of DeepCNN leaf based image classification. The RPNN, FCNN, and scatter plot of RPNN + FCNN impact of each hyperparameter on the performance of different models is shown in Figure 2. The best accuracy of the RPNN model was 88% when the radial basis width (σ) was between 0.5 and 1.0. The 6-layer FCNN architecture achieved an accuracy of 91% without overfitting. The optimal hyperparameters of hybrid model is listed in Table 7.



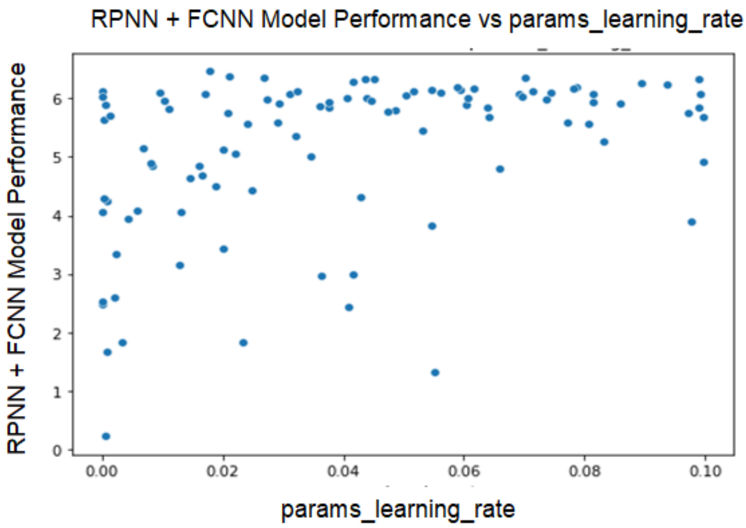


Figure 2. Impact of hyperparameters using RPNN, FCNN, and RPNN+ FCNN

Table 7. RPNN + FCNN Optimal Hyperparameters

Method	Parameters	Values
RPNN + FCNN	Radial Basis Width (σ)	0.34
	Number of Centers (C_i)	86
	Number of Layers (L)	4
	Number of Neurons per layer (N_l)	128
	Activation Function (τ)	ReLu
	Learning Rate (η)	0.063
	Optimizer (o)	Adam
	Batch Size (B)	128
	Dropout (λ)	0.1

TIME COMPLEXITY

To reduce the execution complexity of the hybrid model (RPNN + FCNN), we built a GPU-based acceleration strategy to parallelize the computation of RPNN and FCNN. Both RPNN and FCNN computations are performed on a GPU framework to optimize matrix and kernel computations for parallel execution. As a result, we achieve 4x and 24x computation speedup compared to the standalone RPNN and FCNN techniques, respectively. Computational time complexity and running time of different models is shown in Table 8.

Table 8. Computational time complexity

Models	Time Complexity	Total Running time (S)
RPNN	$O(X.C_i.d + K.C_i)$	8
FCNN	$O\left(X.\sum_{l=1}^{L} W_l\right)$	49
CNN	$O(X.F.H.W.S^2)$	92
DCNN	$O\left(X.\sum_{l=1}^{L} F_l.H_l.W_l.S_l^2\right)$	147
Hybrid model (RPNN+FCNN) with GPU	$O(X.C_i.d + K.C_i) + O\left(X.\sum_{l=1}^{L} W_l\right)$	2

Where, X = Input data points, C_i = RBF centers, d= Feature dimensions, K = Number of clusters, L = Layers, W_l = Weight per layer, F= Kernel size, H = Height of the feature map, W = Width of the feature map, and S = Stride.

Table 9 provides a detailed comparison of various DL models including 2D-CNN-BidLSTM/GRU [13], Generative Adversarial Nets [14], 3D-CNN [15], and DCNN [16]. The proposed DeepCNNMed outperforms existing models in classification accuracy is shown in Figure 3 and requires less computation time.

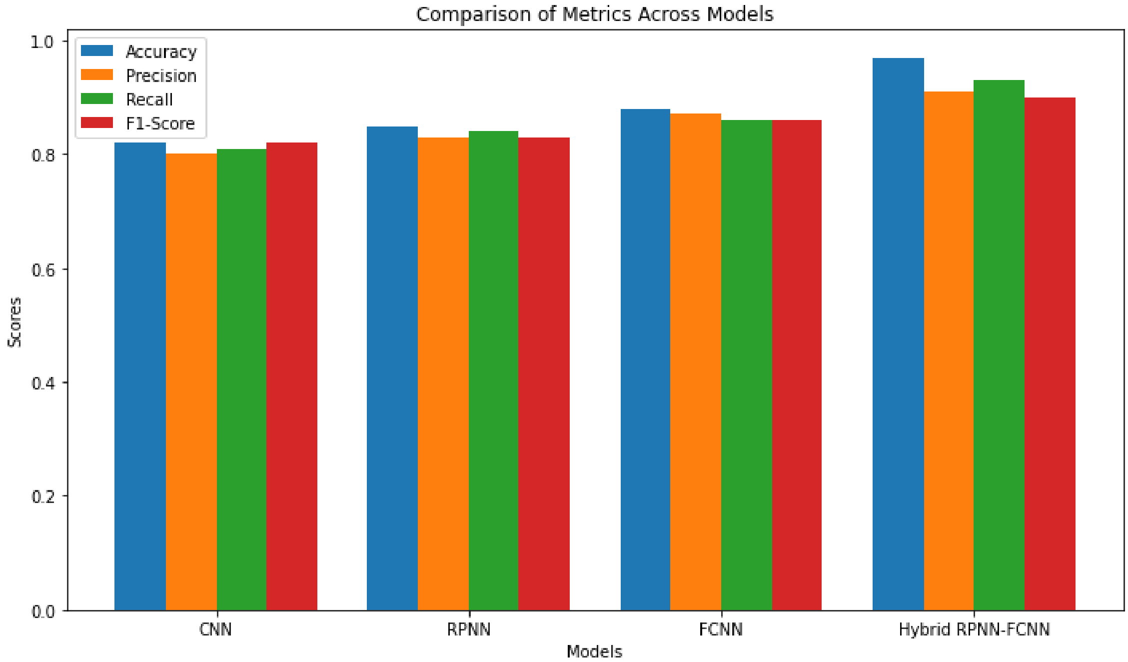


Figure 3. Performance comparisons

CONCLUSION

This work suggests an automated approach for classifying plant species based on their leaves. This task is carried out by deep convolutional neural networks, which increase accuracy. The proposed hybrid architecture also addresses key issues related to interoperability, scalability and regulatory compliance. The model can efficiently process larger datasets due to scalability, and it is achieved by methods such as incremental learning, kernel optimization and parallel computing. Exporting the model in the standard Open Neural Network Exchange (ONNX) format ensures interoperability and enables integration into different platforms and applications. In order to maintain legitimate and interpretable forecasts, regulatory compliance is maintained by introducing privacy-preserving mechanisms into operation and implementing data protection standards. These enhancements make the model not only accurate but also adaptable for large-scale real-world applications. The proposed DeepCNN classifier learns plant features, such as leaf categorization, using hidden layers such as RPNN features and fully connected layers of FCNN. The model learns traits from a Swedish leaf dataset containing 30 plant classes and uses this knowledge to predict appropriate categories of unknown plants with 97.2% accuracy and minimal error. The result is better than the previous work where the highest accuracy was rated at 96.25%.

Table 9. Performance comparisons of state-of-art techniques

Techniques used	Performance metrics	
	F1 score	**Accuracy (%)**
Proposed model (RPNN + FCNN)	0.90	97.2
2D-CNN-BidLSTM/GRU [13]	0.75	73
Generative Adversarial Nets [14]	0.82	96.25
3D-CNN [15]	0.87	95.76
DCNN[16]	0.86	85

LIMITATIONS AND FUTURE SCOPE

The proposed DeepCNN (RPNN+FCNN) efficiently identifies plant species by using grid search and adaptation strategies to determine the optimal kernel bandwidth (σ) and hyperparameters. Experiments with improved activation functions have improved gradient flow, model performance, and prediction accuracy. Increasing the number of hidden layers in FCNN allows learning more complex feature representations, which further improves prediction accuracy. Prediction accuracy can be enhanced by extracting relevant features from RPNN and applying transfer learning to FCNN. In future work, integrating samples from other datasets with similar characteristics to the Swedish Leaf dataset will improve the generalization ability of the model. Furthermore, we will explore ensemble learning and attention mechanisms to reduce variance and improve model robustness.

Funding: No funding was received for this article.
Conflicts of interest: No conflicts of interest in this work.
Ethical Statement: There are no human or animal experiments in this research article.

Data Availability: The data that support the findings of this study are openly available Swedish leaf dataset. https://www.cvl.isy.liu.se/en/research/datasets/swedish-leaf.

REFERENCES

1. Mahajan S, Raina A, Gao XZ, Kant Pandit A. Plant recognition using morphological feature extraction and transfer learning over SVM and AdaBoost. Symmetry. 2021 Feb 22;13(2):356-72.
2. Zhang Y, Cui J, Wang Z, Kang J, Min Y. Leaf image recognition based on bag of features. Appl.Sci.. 2020 Jul 28;10(15):5177-93.
3. Azlah MA, Chua LS, Rahmad FR, Abdullah FI, Wan Alwi SR. Review on techniques for plant leaf classification and recognition. Computers. 2019 Oct 21;8(4):77-94.
4. Kumar M, Gupta S, Gao XZ, Singh A. Plant species recognition using morphological features and adaptive boosting methodology. IEEE Access. 2019 Nov 7;7:163912-8.
5. Jeon WS, Rhee SY. Plant leaf recognition using a convolution neural network. Int.J.Fuzzy Log.Intell.Syst.. 2017 Mar 31;17(1):26-34.
6. Pankaja K, Suma V. Leaf recognition and classification using GLCM and hierarchical centroid based technique. In2018 International Conference on Inventive Research in Computing Applications (ICIRCA) 2018 Jul 11 (pp. 1190-1194). IEEE.
7. Kumar S. Leaf Color, Area and Edge features based approach for Identification of Indian Medicinal Plants”. Int.J.Comput.Sci.Eng.. 2012 Jun;3(3):436-42.
8. Kadir A, Nugroho LE, Susanto A, Santosa PI. Performance improvement of leaf identification system using principal component analysis. Int.J.Adv.Sci.Technol.. 2012 Jul;44(2):113-23.
9. Priya CA, Balasaravanan T, Thanamani AS. An efficient leaf recognition algorithm for plant classification using support vector machine. In International conference on pattern recognition, informatics and medical engineering (PRIME-2012) 2012 Mar 21 (pp. 428-432). IEEE.
10. Anami BS, Nandyal SS, Govardhan A. A combined color, texture and edge features based approach for identification and classification of indian medicinal plants. Int.J.Comput.Appl.. 2010 Sep;6(12):45-51.
11. Sambhaji ES, Andore DB. Leaf recognition algorithm using neural network based image processing. Asian J. Eng. Technol. Innov. 2014;2(2):10-6.
12. Sun Y, Liu Y, Wang G, Zhang H. Deep learning for plant identification in natural environment. Comput.Intell.Neurosci. 2017;2017(1):7361042-48.
13. Jin X, Jie L, Wang S, Qi HJ, Li SW. Classifying wheat hyperspectral pixels of healthy heads and Fusarium head blight disease using a deep neural network in the wild field. Remote Sens. 2018 Mar 4;10(3):395-15.
14. Wang D, Vinson R, Holmes M, Seibel G, Bechar A, Nof S, et al. Early detection of tomato spotted wilt virus by hyperspectral imaging and outlier removal auxiliary classifier generative adversarial nets (OR-AC-GAN). Sci.Rep. 2019 Mar 13;9(1):4377-91.
15. Nagasubramanian K, Jones S, Singh AK, Singh A, Ganapathysubramanian B, Sarkar S. Explaining hyperspectral imaging based plant disease identification: 3D CNN and saliency maps. arXiv preprint arXiv:1804.08831. 2018 Apr 24.
16. Zhang X, Han L, Dong Y, Shi Y, Huang W, Han L, et al. A deep learning-based approach for automated yellow rust disease detection from high-resolution hyperspectral UAV images. Remote Sens. 2019 Jun 29;11(13):1554-70.
17. Azadnia R, Jahanbakhshi A, Rashidi S, Bazyar P. Developing an automated monitoring system for fast and accurate prediction of soil texture using an image-based deep learning network and machine vision system. Meas.. 2022 Feb 28;190:110669-85.
18. Singh LK, Pooja, Garg H, Khanna M. Deep learning system applicability for rapid glaucoma prediction from fundus images across various data sets. Evol.Syst. 2022 Dec;13(6):807-36.
19. Gurusamy R, Seenivasan SR. H2O-Based Stochastic Gradient Descent Grid Search Using Ridge Regression Techniques for Traffic Flow Forecasting. Braz.Arch.Biol.Technol. 2024 May 31;67:e24220670.
20. Khanna M, Singh LK, Thawkar S, Goyal M. Deep learning based computer-aided automatic prediction and grading system for diabetic retinopathy. Multimed.Tools Appl. 2023 Oct;82(25):39255-302.
21. Russel NS, Selvaraj A. Leaf species and disease classification using multiscale parallel deep CNN architecture. Neural Comput.Appl. 2022 Nov;34(21):19217-37.

22. Khanna M, Singh LK, Thawkar S, Goyal M. PlaNet: a robust deep convolutional neural network model for plant leaves disease recognition. Multimed.Tools Appl. 2024 Jan;83(2):4465-517.
23. Dey B, Ferdous J, Ahmed R, Hossain J. Assessing deep convolutional neural network models and their comparative performance for automated medicinal plant identification from leaf images. Heliyon. 2024 Jan 15;10(1):e23655.
24. Pushpa BR, Rani NS, Chandrajith M, Manohar N, Nair SS. On the importance of integrating convolution features for Indian medicinal plant species classification using hierarchical machine learning approach. Ecol.Inform.. 2024 Jul 1;81:102611.
25. Amri E, Gulzar Y, Yeafi A, Jendoubi S, Dhawi F, Mir MS. Advancing automatic plant classification system in Saudi Arabia: introducing a novel dataset and ensemble deep learning approach. Model.Earth Syst.Environ. 2024 Apr;10(2):2693-709.
26. Azadnia R, Al-Amidi MM, Mohammadi H, Cifci MA, Daryab A, Cavallo E. An AI based approach for medicinal plant identification using deep CNN based on global average pooling. Agron. 2022 Nov 2;12(11):2723.
27. Azadnia R, Noei-Khodabadi F, Moloudzadeh A, Jahanbakhshi A, Omid M. Medicinal and poisonous plants classification from visual characteristics of leaves using computer vision and deep neural networks. Ecol.Inform. 2024 Sep 1;82:102683.
28. Pushpa BR, Rani NS. Ayur-PlantNet: An unbiased light weight deep convolutional neural network for Indian Ayurvedic plant species classification. J.Appl.Res.Med.Aromat.Plants. 2023 Apr 1;34:100459.
29. Shelke A, Mehendale N. A CNN-based android application for plant leaf classification at remote locations. Neural Comput.Appl. 2023 Jan;35(3):2601-7.
30. Thangaraj R, Pandiyan P, Anandamurugan S, Rajendar S. A deep convolution neural network model based on feature concatenation approach for classification of tomato leaf disease. Multimed.Tools Appl.. 2024 Feb;83(7):18803-27.
31. Othman NA, Damanhuri NS, Ali NM, Meng BC, Abd Samat AA. Plant leaf classification using convolutional neural network. In2022 8th International Conference on Control, Decision and Information Technologies (CoDIT) 2022 May 17 (Vol. 1, pp. 1043-1048). IEEE.