Multimedia Tools and Applications (2024) 83:33823–33853
https://doi.org/10.1007/s11042-023-16733-8

CNN-based medicinal plant identification and classification using optimized SVM

Himanshu Kumar Diwedi[1] · Anuradha Misra[1] · Amod Kumar Tiwari[2]

Received: 13 March 2023 / Revised: 20 July 2023 / Accepted: 31 August 2023 /
Published online: 25 September 2023

Abstract

The exact and unfailing categorization of medicinal plants exceeds the capabilities of the average individual because it necessitates in-depth subject expertise and physical detection is cumbersome and imprecise owing to human mistakes. There have been multiple efforts to automate the recognition of medicinal plants using images of plant parts like flowers, leaves, and bark. The most trustworthy data source, according to research, is Leaf. An Enhanced Convolutional Neural Network architecture (using modified ResNet50) with Progressive Transfer Learning (ECNN-PTL) has been proposed in this paper. The suggested method uses an improved ReNet50 framework for feature extraction along with PTL. Classification has been done using an Optimized Support Vector Machine (OSVM). The classical SVM hyperparameters are tuned further by the Adam optimizer to achieve a better performance model. During the first stage of training, the initial levels of the pre-trained ResNet50 architecture have been frozen while the recently introduced levels have been taught using a differentiated learning rate. In the second step, the refined model from the first stage is loaded and trained by restructuring. This technique has been replicated so that in these two learning steps, the image size is allowed to gradually rise from 64, 128, and 150 to 256 pixels. The proposed ResNet-50 effectively max-pools the activation from the previous fully connected layer to the subsequent convolution layer. In the trials, the maximum and average activations from the previous convolution are kept, giving the model knowledge of both the approaches and enhancing performance. The Indian Medicinal Plants Database (IMPLAD) has been used to compile the list of online medicinal plant species.The improved ResNet50 modelOSVM classifier in the ECNN-PTLapproach has been compared with baseline models like VGG16, VGG19 and ResNet50 in terms of accuracy, precision, recall, error rate and execution time. The modified ResNet50 + OSVM model achieve a testing phase accuracy of 96.8% and a training phase accuracy of 98.5%.

✉ Himanshu Kumar Diwedi
himanshudiwedi01@gmail.com

Anuradha Misra
amisra@lko.amity.edu

Amod Kumar Tiwari
amod.tiwari@recsonbhadra.ac.in

[1] Department of Computer Science and Engineering, Amity School of Engineering and Technology, Amity University Lucknow Campus, Lucknow, India

[2] Department of Computer Science and Engineering, Rajkiya Engineering College, Sonbhadra, Uttar Pradesh, India

Keywords Convolutional neural network · Support vector machine · ResNet50 · Transfer learning · Indian medicinal plants · Classification · IMPLAD

1 Introduction

The foundation of all scientific study is precise species identification, which is also a crucial step in the processes of biological studies in fields like medicine, geology, and genetic sciences. Effective recognition abilities are necessary for a variety of tasks, including researching an area's biological diversity, tracking threatened species populations, assessing how global warming is affecting biodiversity, and taking pest control measures. For example, peasants, environmentalists, pharmacists, biologists, wildlife specialists, the engineering team of environmental organizations, or just for enjoyment for ordinary people, these tasks are necessary [1]. Automation of the job and its viability for laypeople are widely preferred, particularly in light of the ongoing decline in biodiversity and skilled palaeontologists [2].

As a consequence of the rapid advancement and widespread use of pertinent information systems, along with the accessibility of mobile devices like mobile phones and computers, a substantial volume of digital photographs are freely accessible in internet repositories. The iNaturalist (680,000 photographs of 4500 species) and the Zooniverse (1.19 million photographs of 45 species) databases serve as illustrations of the abundance of images that academics and the general public have collected [3]. Additionally, the widespread digitization of paleontological archives has grown to be a top priority for museums all over the globe and has produced huge digital databases. For instance, the iDigBio platform, a federally supported collection of information on museum exhibits, now offers more than 1.75 million geo-tagged photos of samples of photosynthetic organisms [4]. Modernizing picture-based species detection is now possible, thanks to the current developments in Machine Learning (ML) technology and the availability of vast image databases.

The fastest-growing area of modern science is ML which is prevalent in sectors like digital marketing, medical services, production, data security, and intelligent transport systems. The accessibility and convergence of three factors, including (a) quicker and efficient computer hardware, like multithreaded processing units and general-purpose graphics cards; (b) software programs that reap the benefits of these supercomputing frameworks; and (c) virtually unlimited quantities of learning information for a specific challenge, including digital images, computerized files, posts on social media or other information, is the primary cause of this figurative "outburst" of the method.Artificial Intelligence (AI) in the form of ML is capable of carrying out tasks without being particularly taught to do so. Rather, it goes through a procedure known as training where it understands from precedents of the assigned activity. Interpretation is the technique by which the job can be carried out on new information post-training [5]. ML is particularly beneficial for scenarios where the information is challenging to represent mathematically, such as evaluating image and video footage and significantly aids in gathering knowledge from vast amounts of rapidly increasing datasets.

Acquiring knowledge and understanding from digital photos and videos is the focus of the computer engineering discipline known as computer vision. Technologically, it aims to automate activities that the visual system of humans can perform [6]. The feature extraction and categorization processes make up a computer vision ML pipeline. Computer vision methods can now help biologists with activities like plant recognition due to the advancements in science and technology. For the autonomous examination of plant organs like leaves and flowers, several methods have been put into the research [7].

In biology, leaves have always been used—and in some categories, exclusively—to provide key diagnostic features for plant identification. Plant recognition has been done using text-based generic codes that, among other things, utilise leaf characteristics since the beginning of biological research. Because of this, computer vision scientists have been using leaves as a comparison tool to categorize plants [8]. The characteristics that are most frequently employed to differentiate between the leaves of various species include form [9], texture [10], and vein structure [11]. The history of plant identification techniques, however, demonstrates how specialized information can be encoded by experts to a significant extent by the plant identification methods that are currently in use. Researchers characterize several morphological traits that have been pre-defined by biologists using hand engineering methods.For predictive analysis, they search for processes or techniques that can make the best use of the information. The subgroup of attributes that are most crucial for describing leaf information is then justified based on their effectiveness. With other leaf information or image retrieval methodologies, these traits could, however, vary [12].

Biologists classify plants using a variety of plant parts or characteristics, such as leaves, branches, flowers, etc. Leaf classification is favoured since it is nearly always available. Few professionals, rural residents, and native people have a thorough understanding of indigenous plants, and there isn't simple access to any supporting documentation. To encourage organic living and stop the eradication of prospective plants, it is crucial to spread knowledge about herbs to both the general population and academics [13]. The traditional method for recognizing Indian medicinal plants is laborious, time-consuming, and difficult work since it frequently results in wrong identification since there are more than 8000 kinds of herbs in India [14].

Using ML algorithms to identify medicinal plants demonstrate attention to dividing plant photos into their groupings. Because of the similarities across plant classes and within classes, the likelihood of a cluttered background, and fluctuations in numerous characteristics including brightness and colour, classifying plants using digitized leaf photos is difficult [15]. Therefore, it is crucial to create methods and tools that can effectively identify and interpret the characteristics in leaf photographs.Many studies have recently not focused on developing a model or method to quicken the automatic identification of Indian plants and herbs. The common public and other participants in the production of medicinal herbs would gain knowledge more easily due to the computerized classification of medicinal herbs rather than depending on knowledgeable biologists or ayurvedic (an ancient Indian medical practice) practitioners.

1.1 Contributions

The novel main contributions of this work are as follows:

1. An improved ReNet50 framework for feature extraction (leaf of medicinal plants) along with PTL has been used. During the first stage of training, the initial levels of the pre-trained ResNet50 architecture have been frozen while the recently introduced levels have been taught using a differentiated learning rate. In the second step, the refined model from the first stage is loaded and trained by restructuring.
2. PTL has been employed in the two learning steps, and the image size is allowed to gradually rise from 64, 128, and 150 to 256 pixels.
3. Support vector machines (SVM) have been used for classification. To create a higher performance model, the Adam optimizer is used to further tune the SVM hyperparameters.

4. In terms of accuracy, precision, recall, and F-score, the enhanced ResNet50 model with Artificial Neural Network (ANN) and SVM classifiers in the ECNN-PTL technique has been contrasted with baseline models like AlexNet, VGG16, VGG19, and ResNet50.

The suggested model successfully addresses problems like (1) eliminating variation within a single species of leaf, (2) recognizing variation between separate species of the medicinal leaf, and (3) extracting complicated and distinctive attributes. Consequently, improved results in terms of accuracy and execution time have been obtained.

1.2 Organization of paper

The rest of the paper has been organized as follows: Section 2 describes related research on machine learning models and transfer learning for the identification and classification of medicinal plant species. Section 3 provides a novel framework called an Enhanced Convolutional Neural Network architecture (using modified ResNet50) with Progressive Transfer Learning (ECNN-PTL) to identify medicinal plants. Results and discussion have been given in Section 4. Finally, the conclusion, limitations, and scope for further research have been shown in Section 5.

2 Related works

The identification of species from multiple plant components using ML algorithms has been the subject of continuing study by Nazarenko et al., (2016) [16]. Nevertheless, a leaf is thought to be the most accessible and trustworthy source of evidence, resulting in the fact that leaf photographs have frequently been employed to recognize native plants. Geometry, colour, texture, and vein structure are the primary features that may be retrieved from the leaf photographs.

Various authors have employed various segmentation and categorization techniques. For instance, the MostajerKheirkhah et al., (2019) [17] emphasized that the Probabilistic Neural Network's (PNN) feature vector consisted of all of these qualities. On the publicly accessible Flavia database, it identified the plant species from a leaf image with an overall accuracy of 93.75%.

By combining regional binary structures and a Grey Tone Spatial Dependency Matrix (GTSDM), Naeem et al., (2021) [18] were able to extract texture information from photographs of leaves. Stochastic Gradient Descent, k-NN, and Decision Trees obtained 93.9% accuracy in recognizing medicinal herbs when these attributes were categorized using 6 classifiers.

To retrieve textural information from photographs of leaves, wavelet and temporal dimensionality have been suggested by Kurmi et al., (2022) [19]. Since the information in the photographs was primarily concentrated in the low-frequency sub-bands, the wavelet approach tends to neglect these characteristics. The fractal approach tends to uncover patterns that differ geometrically but have the same spatial pattern. To acquire the texture features, the authors suggest combining wavelet and temporal dimensions. On the Flavia dataset, this characteristic vector's back-propagation learning input into an Artificial Neural Network (ANN) produced an accuracy of 92.01%.

Pushpa et al., (2022) [20] employed the geometric features of the leaf pictures using variable thresholding and Unrestrained Hit or Miss transformation to distinguish between soybeans, white beans, and green peas. These characteristics were fed into the Random Forest (RF) classifier, which had an accuracy rate of 94.8%.

A similar study that examined the effectiveness of several classifiers was put up by Kaur (2019) in [21]. To carry out investigations, geometry and colour features were fed into SVM, k-NN, Naive Bayes (NB), and RF. The maximum accuracy, 95.7%, was attained by RF.

Turkoglu &Hanbay (2019) [22], the authors mention another outstanding piece of work in the same domain to detect leaves associated with the same species. They utilized only a portion of the leaf to extract the colouration, vein structure, texture, and geometrical properties rather than using photos of the entire leaf.Subsequently, for every complete image, the attributes derived from these components were merged. With the aid of geometrical remodelling processes, vein characteristics were determined. Gray Level Co-occurrence Matrix (GLCM), which is predicated on the association among neighbourhood pixels in grey-scale images, was used to derive texture properties. Analytical methods were used to obtain colour characteristics.

The focus has turned to a deep learning technique called CNN over the last few decades. Identifying the incoming images and extracting the useful attributes from them, minimizes human involvement. According to Hu et al., (2018) [23], a Multi-Scale Fusion CNN might be used to recognize plants from photographs of their leaves by downsampling the image data into several low-level images and then feeding each one step by step into the suggested method.The traits gathered at diverse levels are then combined to provide all the data necessary to identify the types of plants.

Geetharamani & Pandian (2019) [24] were able to devise a persistent and dependable deep-CNN, a 9-level framework to identify plant pathogens. To enhance the size of the sample and aid CNN in learning, the data collected was first reinforced. Its accuracy was assessed against many classifiers and it came out at 95.7%.

On the LeafSnap database, Bodhwani et al., (2019) [25] created a 50-level recurrent neural network that achieved an accuracy of 98.7% with an error rate of 0.0514%. Transfer Learning (TL) TL is one of the most dynamic aspects of ML.The idea of applying the information acquired from one job to another is known as TL.

The concept of TL has been clearly outlined in Cheplygina et al., (2019) [26]. The most widely used models for computer vision applications are ResNet50, Inception V3, VGG-16, and VGG-19. These modern structures are open-sourced.

Several academics can utilize the characteristics and values acquired during their training to address issues in their fields. In Wei Tan et al., (2018) [27], it was suggested to use a CNN model named D-Leaf to extract information from the input photos. Its effectiveness was compared against that of refined Alexnet and edge detection from Alexnet. Then, different classifiers received these characteristics as input. The highest accuracy of 95.1% was attained by ANN.

Another example of TL was shown in Kanda et al., (2021) [28], in which the researchers extracted characteristics from leaf photos in the Flavia and LeafSnap databases before classifying them using a logistic regression classification approach. They did this by utilizing a pre-trained model named MobileNet and achieved 91.2% accuracy for the LeafSnap database which contains 185 groups. 99.58% accuracy had been obtained for the Flavia dataset, which has 32 groups.

On the LifeCLEF 2015 database, which contains images of leaves, flowers, branches, fruits, and stems from all categories, Taheri-Garavand et al., (2021) [29] examined the effectiveness of GoogleNet, AlexNet, and VGGNet to evaluate the various characteristics used in ML. While building these models from scratch and then fine-tuning them, they examined the effects of network characteristics like epochs, batch size, and upsampling.

They demonstrated that training AlexNet from scratch produces superior outcomes than fine-tuning GoogleNet and VGG because of its simpler framework. Because of its depthwise detachable convolutions, MobileNet is comparably lighter than other pre-trained algorithms, which reduces the cost of processing. Using their database of medicinal leaves, they have built 16 alternative combinations based on various MobileNet model parameters. The highest reported accuracy of the model was 99.1%.

Sachar et al., (2021) [30] applied transfer learning to compare the feature extraction capabilities of VGG-16, Xception, MobileNetV2 Convolutional Neural Network (CNN) and DenseNet121 architectures to freely available Swedish, Flavia and MalayaKew leaf image datasets. Random Forest is used as classifier to identify the species of given leaf. The evaluations and comparisons of the specified feature extractor models are provided. DenseNet121 achieved maximum accuracy of 100%, 99% and 92.4% respectively in the three datasets.

Kaya et al., (2023) [31] developed a novel approach based on DL for plant disease detection by fusing RGB and segmented images. A multi-headed DenseNet-based architecture was developed, considering two images as input. They evaluated the model on a public dataset, PlantVillage, consisting of 54,183 images with 38 classes. The fivefold cross-validation technique achieved an average accuracy, recall, precision, and F1-score of 98.17%, 98.17%, 98.16%, and 98.12%, respectively. The approach distinguished various plant diseases with different characteristics by image fusion. The high success rate with low standard deviation proves the robustness of the model, and the model can be integrated into plant disease detection and early warning system.

Keceli et al., (2022) [32] developed a novel approach based on the multi-task learning strategy, using shared representations between these related tasks, because they perform better than individual models. They applied a multi-input network that uses raw images and transferred deep features extracted from a pre-trained deep model to predict each plant's type and disease. They developed an end-to-end multi-task model that carries out more than one learning task at a time and combines the Convolutional Neural Network (CNN) features and transferred features. They then evaluated the model using public datasets. The results of their experiments demonstrated that this Multi-Input Multi-Task Neural Network model increases efficiency and yields faster learning for similar detection tasks.

Keceli et al., (2022) developed a deep learning model based on CNN to recognize hand gestures improving recognition rate, training, and test time. The approach included data augmentation to boost training. Furthermore, five popular deep learning models are used for transfer learning, namely VGG16, VGG19, ResNet50, DenseNet121, and InceptionV3 and compared their results. These models are applied to recognize 10 different hand gestures for near-infrared images and 24 ASL hand gestures for colored natural images. The developed CNN model, VGG16, VGG19, Resnet50, DenseNet121, and InceptionV3 models achieve recognition rates of 99.98%, 100%, 99.99%, 91.63%, 82.42% and 81.84%, respectively on near-infrared images. For colored natural ASL images, the models achieve recognition rates of 99.91%, 99.31%, 98.67%, 91.97%, 93.37%, and 93.21%, respectively. The developed model achieved promising results spending the least amount of time.

Chen et al., (2022) [33] made a first comprehensive evaluation of deep transfer learning (DTL) for identifying common weed species specific to cotton (*Gossypium hirsutum* L.) production systems in southern United States (U.S.). A new dataset for weed identification was created, consisting of 5187 color images of 15 weed classes collected under natural light conditions and at varied weed growth stages, in cotton fields (primarily in Mississippi and North Carolina) during the 2020 and 2021 growth seasons. They evaluated 35 state-of-the-art deep learning models through transfer learning with repeated holdout validations and established an extensive benchmark for the considered weed identification task. DTL achieved high classification accuracy of F1 scores exceeding 95%, requiring reasonably short training time (less than 2.5 h) across models. ResNeXt101 achieved the best overall F1-score of 98.93 ± 0.34%, whereas 10 out of the 35 models achieved F1 scores near or above 98.0%. However, the performance on minority weed classes with few training samples was less satisfactory for models trained with a conventional, unweighted cross entropy loss function. A weighted cross entropy loss function was adopted, which achieved substantially improved accuracies for minority weed classes (e.g., the F1-scores for Xception and MnasNet on the Spurred Anoda weed increased from 48% to 90% and 50% to 82%, respectively). Furthermore, a deep learning-based cosine similarity metric was employed to analyze the similarity among weed classes, assisting in the interpretation of classifications.

Huang et al., (2022) [34] established a tomato pest image dataset of common tomato pests. Deep learning (DL) convolutional neural network (CNN) models were applied to classify eight categories of tomato pests. Transfer learning was used to reduce training time. The DL models were used to extract features, and the extracted pest features were combined with three machine learning (ML) classifiers, including discriminant analysis (DA), support vector machine, and k-nearest neighbor method (KNN). Hyper-parameters were automatically optimized through Bayesian optimization. After image augmentation, VGG16 model exhibited the highest performance than the other models, with an accuracy of 94.95%. Regarding the CNN + ML models, the ResNet50 with discriminant analysis model achieved 97.12% classification accuracy. Combining the advantages of DL and ML could not only simplify the feature extraction process and reduce the training time, but also provided experts and farmers with effective and immediate assistance in identifying pests, which helped reduce economic and crop yield loss.

Overall, a variety of methods have been applied to the categorization of Indian medicinal plants, but there is still room for improvement. Accuracy and execution time are subject to trade-offs. Certain works have excellent accuracy but need a lot of computing, while others have excellent processing but poor accuracy. The research suggested in this study fills the gap between the two and offers a computationally simple model (reduced execution time) without sacrificing its accuracy. These studies encourage the development of a unique model that extracts leaf properties using a modified CNN architecture. The suggested scheme will significantly advance the area of automated medicinal plant classification.

3 Enhanced convolutional neural network architecture (using modified ResNet50) with progressive transfer learning (ECNN-PTL)

In our proposed system, the main objective is to achieve automatic classification of medicinal plants using an enhanced CNN architecture, specifically a modified ResNet50 model. The IMPLAD database [35] has been utilized to compile a collection of online medicinal plant species. However, conventional CNN models trained on this dataset have shown decreased accuracy. To address this issue, we have developed a modified ResNet50 model-based CNN architecture that improves accuracy while reducing execution speed. Additionally, we extend the Transfer learning section by incorporating insights from related research, such as studies [32] to [35], which further enhance our understanding and application of transfer learning techniques in the context of medicinal plant classification.

The automatic classification of medicinal plants using enhanced CNN (with modified ResNet50) architecture and PTL approach is the main objective of this paper. The collection of online medicinal plant species has been compiled using the IMPLAD database [35]. Accuracy is decreased when using this dataset to conventionally create a CNN model. Therefore, a modified ResNet50 model-based CNN architecture has been developed which provides better accuracy with reduced execution speed.

Image pre-processing, segmentation, extraction of features, and classification are the four stages of the proposed work. The first step is to capture digital pictures of the plant specimens. The segmentation and pre-processing phases receive the leaf pictures from the medicinal plant for further extraction of features and classification. Using the updated ResNet50 architecture and PTL, the leaf's attributes are extracted to acquire significant data. Finally, the attributes are categorized using SVM and OSVM algorithms.

3.1 Dataset description

The IMPLAD (Indian Medicinal Plants Database) dataset [35] has regularly updated botanical information (plant names and their related words, 250,000 vernacular identities in 30 Indian languages, availability, state inventory level, geographical locations, plant photographs), as well as cultural heritage (Vedic labels, bibliographies from 32 major ancient literature of Indian systems of medicine, Naturopathy, Herbal pharmaceutics and herbal extract information) of 6500 medicinal plants available in India.

The central part or the IMPLAD is the taxonomy details of plant species. The international code of botanical nomenclature standards is one protocol followed in this aspect. However, it is not strictly followed at this point of time due to fear of repeating the same exercises followed elsewhere. The number of levels represented in one taxa is limited to family, genus, species, varieties, etc. The information from other international databases also referred to confirm the authenticity of existing taxonomical nomenclature of plant species recorded in the IMPLAD.

Plants are known by their names and credited with their uses. Polynomial nomenclature system of Indian knowledge systems and different cultures has contributed a new dimension in plant nomenclature. Even though there are few problems with the polynomial system due to its specificity, it was not practically possible to a country like India to communicate with single language and single name for a plant. The knowledge and culture are still diverse, and it holds absolutely logical to follow the documentation of hundreds of names for a plant species. The names are ascribed to a plant for describing its form or habit (svarupabodhaka), revealing its quality (gunabodhaka), action (karma) of the plant. Table 1 shows samples of IMPLAD dataset [32] which includes 40 sample medicinal Plants.

3.2 Preprocessing techniques

This subsection discusses the preprocessing methods utilized for the self-created database (IMPLAD). The IMPLAD dataset is constructed by randomly dividing the photos into a training and testing set using a 70:30 split proportion. During the creation of the database, some unrelated photographs were further acquired from the internet. However, physically screening them for incorrectly labelled or unclear photos is a time-consuming process. To select photographs with the biggest training fault, the system is thus first trained using the initial collection of pictures. Both successfully classified and incorrectly labelled or ambiguous images have the highest training flaw. All of these photos are sent for the most serious categorization process. In the end, the incorrectly labelled or ambiguous photographs are

Table 1 Samples of IMPLAD dataset

S No.	Medicinal Plants	Scientific Name	Description
1	Ashwagandha	*Withania somnifera*	Adaptogenic herb, promotes vitality and reduces stress
2	Tulsi	*Ocimum sanctum*	Immune-modulating herb, supports respiratory health
3	Turmeric	*Curcuma longa*	Anti-inflammatory spice, aids digestion and joint health
4	Amla	*Phyllanthus emblica*	Rich in Vitamin C, boosts immunity and aids digestion
5	Brahmi	*Bacopa monnieri*	Memory-enhancing herb, supports cognitive function
6	Neem	*Azadirachta indica*	Antimicrobial herb, supports skin health and detoxification
7	Giloy	Tinospora cordifolia	Immunomodulatory herb, supports liver and immune health
8	Haritaki	*Terminalia chebula*	Digestive tonic, promotes detoxification and rejuvenation
9	Guggul	Commiphora mukul	Cholesterol-lowering resin, supports joint and heart health
10	Shatavari	Asparagus racemosus	Female tonic, supports hormonal balance and reproductive health
11	Manjistha	Rubia cordifolia	Blood purifier, supports skin health and detoxification
12	Kutaja	*Holarrhena antidysenterica*	Antidiarrheal herb, supports digestive health
13	Guduchi	Tinospora cordifolia	Immunomodulatory herb, supports liver and immune health
14	Bhringraj	*Eclipta alba*	Hair tonic, promotes hair growth and scalp health
15	Arjuna	Terminalia arjuna	Cardiovascular herb, supports heart health and circulation
16	Amalaki	*Phyllanthus emblica*	Rejuvenating herb, supports digestion and skin health
17	Vacha	*Acorus calamus*	Brain tonic, enhances memory and cognitive function
18	Bibhitaki	*Terminalia bellirica*	Digestive tonic, supports detoxification and respiratory health
19	Nagarmotha	*Cyperus rotundus*	Digestive herb, supports gut health and metabolism
20	Shankhpushpi	Convolvulus pluricaulis	Nervine tonic, supports brain health and memory
21	Jatamansi	Nardostachys jatamansi	Calming herb, supports stress relief and sleep quality
22	Kapikacchu	*Mucuna pruriens*	Aphrodisiac herb, supports reproductive health and libido
23	Triphala	A combination of three fruits: Amalaki, Bibhitaki, Haritaki	Digestive tonic, supports detoxification and bowel regularity
24	Musta	*Cyperus rotundus*	Digestive herb, supports gut health and reduces inflammation

Table 1 (continued)

S No.	Medicinal Plants	Scientific Name	Description
25	Yashtimadhu	*Glycyrrhiza glabra*	Demulcent herb, soothes throat and digestive irritation
26	Vasa	*Adhatoda vasica*	Respiratory herb, supports respiratory health and cough relief
27	Pippali	*Piper longum*	Digestive and respiratory herb, supports metabolism and lung health
28	Gokshura	*Tribulus terrestris*	Kidney tonic, supports urinary tract and reproductive health
29	Shankhpushpi	*Clitoria ternatea*	Brain tonic, enhances memory and cognitive function
30	Brahmi	*Centella asiatica*	Nervine tonic, supports brain health and memory
31	Haridra	*Curcuma longa*	Antioxidant spice, supports liver health and skin radiance
32	Shankapushpi	Convolvulus pluricaulis	Nervine tonic, supports brain health and memory
33	Bael	*Aegle marmelos*	Digestive herb, supports gut health and digestion
34	Devdaru	*Cedrus deodara*	Respiratory herb, supports lung health and congestion relief
35	Kalmegh	*Andrographis paniculata*	Immune-supportive herb, supports liver health and detoxification
36	Vrikshamla	Garcinia cambogia	Weight management herb, supports appetite control and metabolism
37	Chandan	*Santalum album*	Cooling herb, supports skin health and reduces inflammation
38	*Aloe Vera*	*Aloe barbadensis*	Healing succulent, supports skin health and wound healing
39	Bhumi Amla	*Phyllanthus amarus*	Liver tonic, supports liver detoxification and digestive health
40	Shankhpushpi	*Clitoria ternatea*	Brain tonic, enhances memory and cognitive function

manually rectified (in the situation of incorrectly labelled categorization) or removed (in the event of ambiguous pictures).

Once the database is short and unbalanced, the key issue that may arise is that the majority class (the one with more information) is strongly prejudiced, and the system will be eager to forecast everything as the majority class. Data enhancement is a method for enlarging a database by creating virtual replicas of each picture using different image modification techniques, such as mirroring, twisting, shearing, cutting, magnifying, and adjusting colour saturation. Using a data enrichment approach known as “oversampling” and several distinct sets of modifications, the current dataset is preprocessed. Algorithm 1 details the technique utilized for oversampling the database.

A list of five enhancement techniques is created by concatenating a series of random changes for each input picture. A randomized unique picture is taken from the database, and only one enhancement method $\{s_x\}$ is randomly picked per picture and conducted with a randomized amplitude within preset thresholds, where the likelihood condition to apply a modification is determined by the variable p. The use of oversampling methods on the photos of the IMPLAD produced a balanced database in which the number of images in each class was proportional.

In the benchmark samples, the processes required to detect and rectify mislabeled or vague images are not executed. The research has implemented the recommended oversampling approaches to address the data imbalance issue. The oversampled domain sample set currently has 183 images per class.

3.3 Image segmentation

To distinguish the foreground and background, the leaf image of a medicinal plant that was photographed or scanned is scaled to 1600 × 1100 resolution. The source file’s black backdrop is replaced with a white one using a variety of OpenCV approaches. An estimated contour, foreground and backdrop masking have been made first, then a grab-cut approach is used to determine the certain backdrop and forefront of the image. Filtering has been applied to reduce salt and pepper noise. Using the required modules for Python language, the entire procedure has been digitalized. Algorithm 2 discusses the specific procedures taken to segment the foreground and backdrop of the concerned medicinal plant leaf using Python libraries like Open CV2, Python Image Library (PIL), and NumPy.

3.4 Modified ResNet50 structure

The suggested Convolutional Neural Network (CNN) design comprises a pre-trained ResNet50 structure, known as the “body,” and customizable layer groupings appended at the end, known as the “header.” The final branch of the initial ResNet model is re-architected using Variable Concat Pool (VCP), Variable Average Pool (VAP), Variable Maximum Pool (VMP), and fully connected levels accompanied by a mix of flattened, Batch Normalization (BN), dropouts, and linear levels.

As shown in Fig. 1, the updated ResNet50 structure has six levels, comprising layers for input and output. The original ResNet-50 architecture consists of six levels (Input, level-1, level-2, level-3, level-4, output), and Fully Connected (FC) levels. By adding a “Header” layer made up of Variable Average Pool (VAP) and Variable Max Pool (VMP) layers concatenated by Variable Concat Pool (VCP), the suggested approach re-architects the conventional FC levels. The input image undergoes preliminary convolution using a 6 × 6 kernel

Algorithm 1 Oversampling database classification method

```
Input
Input leaf image
Initialisation:
                                        $S = \{s_1, s_2, s_3, s_4, and\ s_5\}$
$s_1$=central plant + shifting scale & rotate + CLAHE
$s_2$= $randomised\ rotate\ (90^{\circ}) + shifting\ scale\&rotate$
$s_3$= $flipping + resizing + randomised\ brightness$
$s_4 = longest\ maximum\ size + transposes_1$
$s_5 = strongest\ transformation$
Output:
 Classified leaf images
    M = Target size of the picture in a group
     For every group g
 P = number of pictures in a group
          While $P < M$ do
               Select randomised picture k
 Perform randomised transformation from S
 Obtain results $(o)$
 Upgrade group $g = \{g, o\}$
 Go to the next leaf picture
 End
```

size, which is proceeded by a max-pool level with a 3×3 kernel size and a stride of 2 (s = 2). With this specific process, the image dimensions are each reduced by one-fourth while the band size is expanded to 64. Every level starts with downsampling or an identification block, and levels 1, 2, 3, and 4 each feature two residual connections with three layers. P_1 and P_2 are the two pathways in the down-sampling segment. Triple convolution layer with kernel dimensions of 1×1, 3×3, and 1×1 have been gathered and stored in Path P_1, one at a time. The incoming image's length and width are accurately reduced by

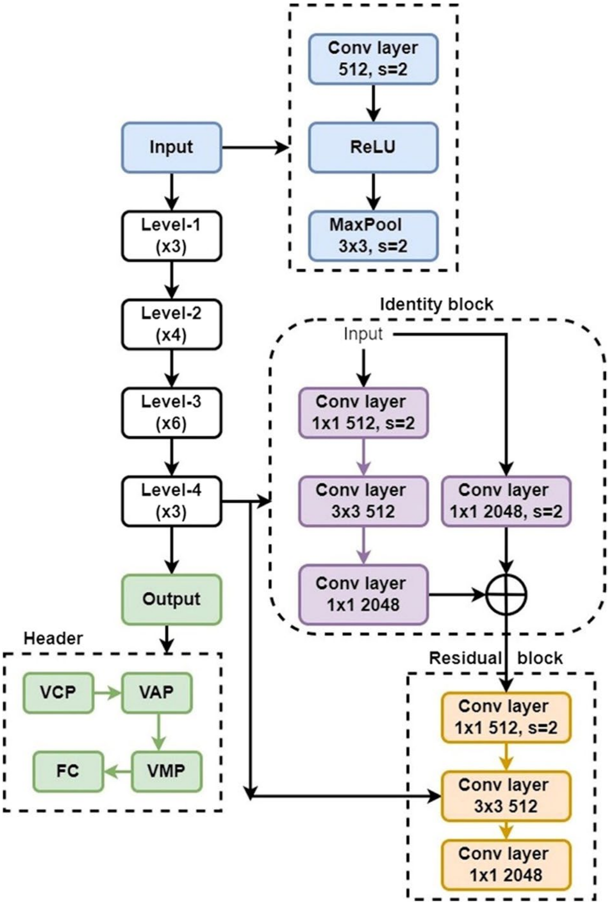
Algorithm 2 Medicinal Leaf Image Segmentation

```
Input: Let $I$ be the input leaf image with a lighter backdrop
Output: Obtain a new leaf image, $L_{segmentted,}$ placed on a white backdrop
Step 1: Resize the input leaf image, $I$, to a target size of $1600 \times 1100$ pixels.
Step 2: Apply Gaussian blur to the image, $I$, using a kernel G to obtain the smoothed image, $I_{smoothed} = I * G$, where $*$ denotes the convolution operation.
Step 3: Employ a hierarchical edge detection algorithm, such as Canny edge detection, to identify the boundaries of the leaf in the smoothed image, $I_{smoothed}$ . Apply a filter $F$, to remove salt and pepper noise, resulting in the denoised edge-detected image, $I_{edges} = Canny(I_{smoothed}) * F$.
Step 4: Trace the outline of the leaf in $I_{edges}$ to obtain the leaf boundary, $B$. Calculate the areas of the foreground , $A_{foreground}$, and backdrop, $A_{backdrop}$, regions based on the leaf boundary $B$.
Step 5: Utilize the GrabCut or GrabSlice image segmentation method with the initial foreground region set to $A_{foreground}$vand the initial background region set to $A_{backdrop}$ to separate the leaf from the backdrop. Obtain the segmented leaf image, $L_{segmentted}$ , using the segmentation result.
Step 6: Place the segmented leaf image, $L_{segmentted}$ , on a white backdrop to obtain the final output
```

half by the initial convolution layer using a step of 2, and the bandwidth is doubled by the following two levels. To accomplish output aggregation between Path P_1 and Path P_2, Path P_2 is used to transform the input structure into path P_1's output structure. As a result, the system ultimately has a mean Max - pooling followed by a level with 500 neurons that are completely linked. The issue of disappearing gradients is addressed using a deep residual structure, in which a constraint design is used to reduce the communication overhead when more levels are added to the network design. Each unit has 11 convolutions levels added to the start and end, as seen within the identity unit of Fig. 1. It aids in reducing the number of variables without deteriorating and impairing system performance.

The architecture of modified ResNet50 with newly added levels implemented in the proposed ECNN-PTL approach is given in Fig. 2. The system has body, header, and layer parts. The purpose of maintaining both maximal and mean activations in the final convolution tier is to enable the neural network to determine the optimal strategy without requiring individual experimentation. The result of the final layer's H x W feature mapping

Fig. 1 Architecture of the modified ResNet50 model



outperforms the mean results and vice versa. The redesigned design combines the VAP and VMP layers with the VCP level. ResNet50 simple max-pools the activating from the last convolutional level to the following fully connected level. As a transition phase, three separate pooling levels are employed to link the convolutional level to the fully connected levels. The maximum and average activations are maintained from the previous convolution, which provides the model with knowledge about approaches and enhances its efficiency. In adaptive accumulation, the stride (*S*) and kernel size (K) are chosen dynamically based on the specified output channels. Equations (1) and (2) are used to determine the variable's stride and kernel size based on the picture's result dimension.

$$S = \frac{N_p}{N_q} \tag{1}$$

$$K = N_p - \left(N_q - 1\right)D_{sp} = 0 \tag{2}$$

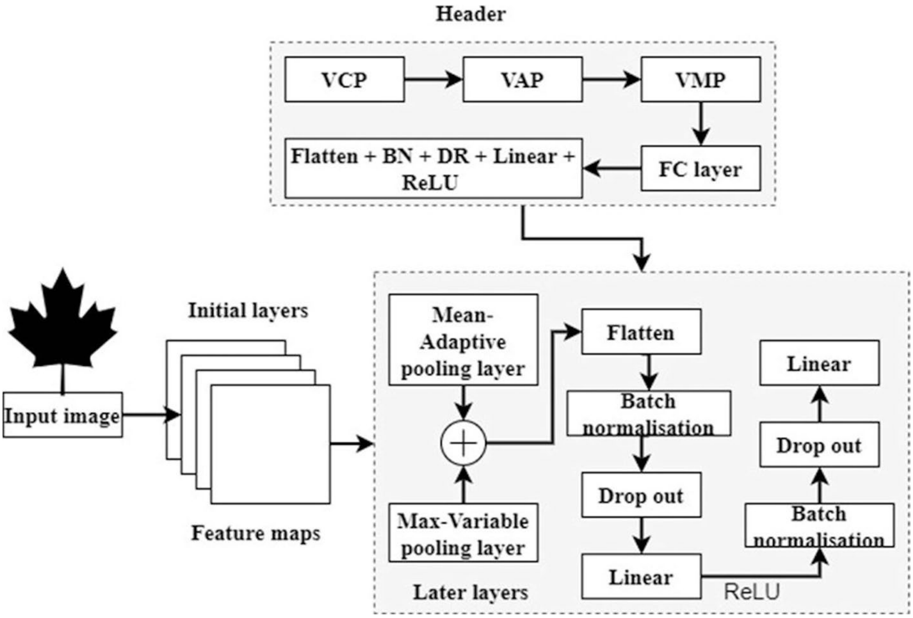


Fig. 2 Architecture of modified ResNet50 with newly added levels implemented in the proposed ECNN-PTL approach

The input sample dimension is denoted as N_p, and the output picture dimension is expressed as N_q. The stride and padding data are indicated as D_{sp}.

The BN level is accountable for normalizing the distribution of both positive and undesirable characteristics produced by the preceding convolutional layer. BN is added either before or after nonlinearities such as sigmoid or Rectified Linear Unit (ReLU). The ReLU activating level will terminate negative characteristics; hence, only positive characteristics will be normalized using BN without biassing them with characteristics that do not advance to the following convolutional level. The output is expressed in Eq. (3).

$$o = BN\{func(Wx + b)\} \tag{3}$$

W is denoted as weight and *b* is expressed bias, which are learned variables, *BN* is batch standardization, and ***func*** is a linear function such as ReLU. The architecture implements a dropout of 0.6 across the BN level and the level preceding ReLU and 0.3 across the final level and ReLU. After experimenting with several pairings of dropouts, such as 0.3/0.5 and 0.4/0.6, which led to overfitting and underfitting, respectively, the research settled on 0.4/0.3 as the optimal dropout for the design.

3.5 Model training and PTL

This subsection explains the approaches used to adjust the model's variables and evaluation metrics. The intelligence of a model is contingent on the procedures used to choose the starting weights, dropouts, feature extraction, batch normalization, etc. All of these variables must correspond to the learning algorithm to minimize the objective functions.

The coefficients of the ResNet50 algorithm are collected from the IMPLAD dataset. Transfer learning is carried out by retraining the model that predicted the likelihood of each category in the leaf database. With the growth of digitalization, constructing a model from scratch and training it at each stage is laborious and time-consuming. Instead of this laborious process, transfer learning provides the benefit of obtaining the weights from previously trained images.

The unique 1000-column weighted vector after the initial ResNet50 structure training on the IMPLAD data set is used. As a result of the distinct categories in the leaf databases, this weighted vector is irrelevant and unimportant to the study and tests. Due to this, two new weighted vectors with a ReLU feature inserted in between them were substituted for the previous weighted vector in the pre-trained system. The parameters in the recently introduced "header" of the redesigned ResNet50 design are initialized. It prevents layer-activating results from inflating or disappearing during the forward pass of CNN. A layer "l" weighted vector is started with randomized values drawn from a typical normal dispersion, with each randomized number magnified by a, where m is the number of incoming networking links (f_{in}) to layer l and the outgoing links are denoted f_{out}. This method initialized the weights by selecting values from a homogenous randomized dispersion limited by the function expressed in Eq. (4).

$$\pm\frac{\sqrt{6}}{\sqrt{f_{in}}+\sqrt{f_{out}}} \tag{4}$$

The incoming and outgoing links are expressed f_{in} and f_{out}. All of the research's training and verification datasets are normalized using the average and standard deviation values obtained from the pre-trained ResNet50 structure. The normalized plant image is indicated in Eq. (5).

$$T_p = p - \frac{\alpha}{\mu} \tag{5}$$

Where, p is the pixel intensity at the link, a and μ are the average and standard deviation for each link (R, G, and B) on pictures from the IMPLAD training dataset. The mean value of RGB is set to [0.42, 0.47, 0.41], and the standard deviation value is set to [0.21, 0.23, 0.24].

The learning speed is used to develop CNN since it represents the pace at which the model's variables are changed. In particular, the variable learning rate improves the efficiency of the system by using transfer learning speeds for specific regions of the CNN. The learning speed finding curve provides a large amount of necessary insight for determining the ideal learning rate. It learns tiny data bunches by gradually increasing the learning speed from an extremely low value (10^{-8}) for $init_{lr}$ to a high value of max_{lr}. The learning speed increases from a lower learning speed (min_{lr}) to a larger learning speed (max_{lr}) and then returns to the original low learning speed. In the last stage, known as the annihilation step, the amount of the training speed is decreased to 1/10th of the original low training speed, min_{lr} As a regularisation parameter, the training speed prevents the system from overfitting at position a. When the training speed enters its cosine-annealing stage, the inertia scheduling decreases to the lowest possible value and subsequently returns to its starting value by following the cosine. The training rate is adjusted after each mini-batch and expressed in Eq. (6), and the maximum iteration is indicated in Eq. (7).

$$lr_x = init_{lr} \times \left(\frac{\max_{lr}}{init_{lr}} \right)^{x/m} \tag{6}$$

$$\max_{lr} = init_{lr} \times y \tag{7}$$

Where m is the number of repetitions and y is the factor that increases the training speed after each mini-batch. The initial iteration is denoted as $init_{lr}$, the maximum iteration is denoted as $\max_{lr}$, and the input is denoted x. All measurements and investigations indicate that the spectrum is between 0.95 and 0.85.

3.6 Fine-tuning of the system

During the first stage of training, the initial "n" levels of the pre-trained ResNet50 system are frozen, meaning the weights are not modified during backpropagation. Nevertheless, the newly inserted level "l" is developed using a variable training speed derived from the one-cycle strategy. The first level of CNN detects basic variations of the lines, the second level recognizes simple forms, and the third level may identify mixtures of lines and patterns while succeeding layers concentrate more on the particular job. It is quite that upgrading variations with a similar learning rate can improve early layer characteristics since the description of a horizontal line anticipated in previous layers will be the same regardless of the dataset. Consequently, the primary levels were not taught during the initial training period.

In the second phase of learning, the refined structure from phase I is loaded, and the entire system is trained by defrosting. In these two rounds of learning, this procedure is replicated in such a manner that the picture's dimension increases from 64,128,150 to 256 pixels through PTL.

3.7 Optimized SVM (OSVM) classification

The SVM is a traditional supervised learning technique that excels at handling larger dimensional spaces and asymmetric attribute values [15]. The largest marginal proximity between the attribute values of the groups to be categorized is what SVM seeks to maintain. This marginal proximity represents the separation from the choice limit that is the greatest. The attribute values that are closest to the separating hyperplane determine where it will be located. Two approaches have been created in this paper using an SVM classifier; one model does not use the Adam optimizer, and the other model uses the Adam optimization algorithm (OSVM) to enhance the accuracy and processing speed of the SVM model parameters (kernel and P values) on the included information. The *S* parameter also referred to as the "price parameter," regulates the trade-off between the accurate categorization of training information points and the flat choice limit.

Cross-entropy losses are used as the error function to forecast the category in the SoftMax tier, and enhancement tactics such as illumination, affine, rotations, warping, and brightness are utilized during the learning phase. An enhanced Adam optimization has been used for optimization. The laborious process of training a system with a set training speed is additionally updated and enhanced by applying the notion of a one-cycle strategy, in which cyclic training speeds are used instead of constant quantities of the training speeds. It greatly aided in achieving higher classification reliability by circularly adjusting the training speed between reasonable bounds rather than progressively lowering it. The system consists of two learning phases. By upgrading the learned variables by backward propagating and implementing the differentiated training speed technique, the loss component is reduced. A binary method's cross-entropy damage is expressed in Eq. (8).

$$CE = -\{q\ log(log\ x) + (1 - q)\log(\log(1 - x))\} \tag{8}$$

The output is denoted q, the input is expressed as x, and the cross entropy is denoted CE. A unique loss is computed and totalled for each category and the result is expressed in Eq. (9).

$$-\sum_{i=0}^{n-1} q_{0,i}\log\left(x_{0,i}\right) \tag{9}$$

where n is the number of categories, $q_{0,\ i}$ represents the real label, and $x_{0,\ i}$ represents the anticipated label. Table 2 shows the Training parameters of the learning system.

4 Results and discussion

The research conducted tests on the custom database by constructing source and target models. The CNN frameworks such as VGG-16, VGG-19, ResNet50 and modified ResNet50 trained on the IMPLAD database along with PTL have been employed for extracting features.The classification has been doneusing a three-level machine learning classification algorithm, namely SVM and OSVM on the IMPLAD database. The classification of the target model is aided by the features collected from the convolutional layers of

Table 2 Training parameters of the learning system

Learning System	Training Parameters
Redesigned ResNet50	Weight initialization, randomized values from a normal distribution
	m: Number of incoming networking links to layer l
	f_{in}: Incoming links to layer l
	f_{out}: Outgoing links from layer l
	α: Average for each link (R, G, B) from IMPLAD training dataset
	μ: Standard deviation for each link (R, G, B) from IMPLAD training dataset
Variable Learning Rate CNN	Learning rate schedule ($\boldsymbol{init}_{lr}$,max_{lr},min_{lr})
	min_{lr}: Lower learning rate
	max_{lr}: Higher learning rate
SVM Classifier	Kernel type, price parameter (S)
	Adam optimizer (OSVM)
Cross-Entropy Loss with Enhanced Adam Optimization	Illumination, affine, rotations, warping, brightness augmentation
	One-cycle strategy for training speed adjustment
	CE: Cross entropy
	n: Number of categories
	$q_{0,\ i}$: Real label
	$x_{0,\ i}$: Anticipated label

the original model utilizing the IMPLAD database. The IMPLAD database is supplied to the targeted model to categorize the forty distinct Indian plants.

Six distinct types of source models are proposed, including VGG16 + SVM, VGG19 + SVM, ResNet50 + SVM, ResNet50 + OSVM, Modified ResNet50 + SVM, and Modified ResNet50 + OSVM. Python was utilized to execute the research using the Keras and TensorFlow machine learning frameworks. The leaf photos in the training dataset are modified by rotation, flipping horizontally and vertically, resizing, and introducing noise. All CNN methods for the IMPLAD dataset are assessed for their precision. The training and testing sets of the IMPLAD database each include 1500 and 500 pictures. All models are executed using Google Colab Pro. Table 3 shows System Configuration.

The outcomes of each model's training and testing accuracy analysis are evaluated, and the cumulative outcomes of all six models are assessed. Equation (10) is used to calculate the accuracy.

$$A = \frac{T_{(p)} + T_{(n)}}{T_{(p)} + T_{(n)} + F_{(p)} + F_{(n)}} \tag{10}$$

The true positive and true negative values are denoted by the notations $T_{(p)}$ *and* $T_{(n)}$, respectively, while the false positive and false negative values are denoted by the notations F $F_{(p)}$ *and* $F_{(n)}$, respectively. The training results are shown in the plot in Fig. 3a, while the testing results are shown in Fig. 3b. When it comes to calculating the detection and classification accuracy during the training phase, all six models have the same level of performance. During the testing phase, the performance of the updated ResNet50 + OSVM model was superior to that of all the other models. When the epoch size is changed from 5 to 60, all models show a greater accuracy level than when they are directly coupled to one another.

Analyses are performed on the average simulation accuracy of all six models using an epoch size that ranges from five to sixty with a step size of five. Using the Sines equation, the accuracy is determined at (10). The results of evaluating the overall models' accuracy in both the training and testing stages are presented in Fig. 4. The average accuracy of all of the models is computed and plotted. In both the training and testing phases, the total simulation results show that the upgraded ResNet50 + OSVM model outperforms the other models. The improved ResNet50 + OSVM model achieved an accuracy of 98.5% during the training phase and 96.8% during the testing phase.

Table 3 System Configuration

System Configuration	Description
Hardware	CPU: Intel Core i7-8700K @ 3.70GHz, GPU: NVIDIA GeForce RTX 2080 Ti, RAM: 16GB DDR4
Software	Operating System: Windows 10, Python 3.7, TensorFlow 2.5, Keras 2.4, OpenCV 4.5, NumPy 1.19
Dataset	IMPLAD database: 1500 training images, 500 testing images
Model Architecture	Modified ResNet50 with Variable Concat Pool (VCP) and Variable Average Pool (VAP)
Preprocessing Techniques	Image scaling, Background replacement, Segmentation, Noise reduction
Transfer Learning	Pre-trained ResNet50 weights on IMPLAD dataset
Classification Methods	Support Vector Machine (SVM), Optimized Support Vector Machine (OSVM)
Optimization Algorithm	Adam optimizer

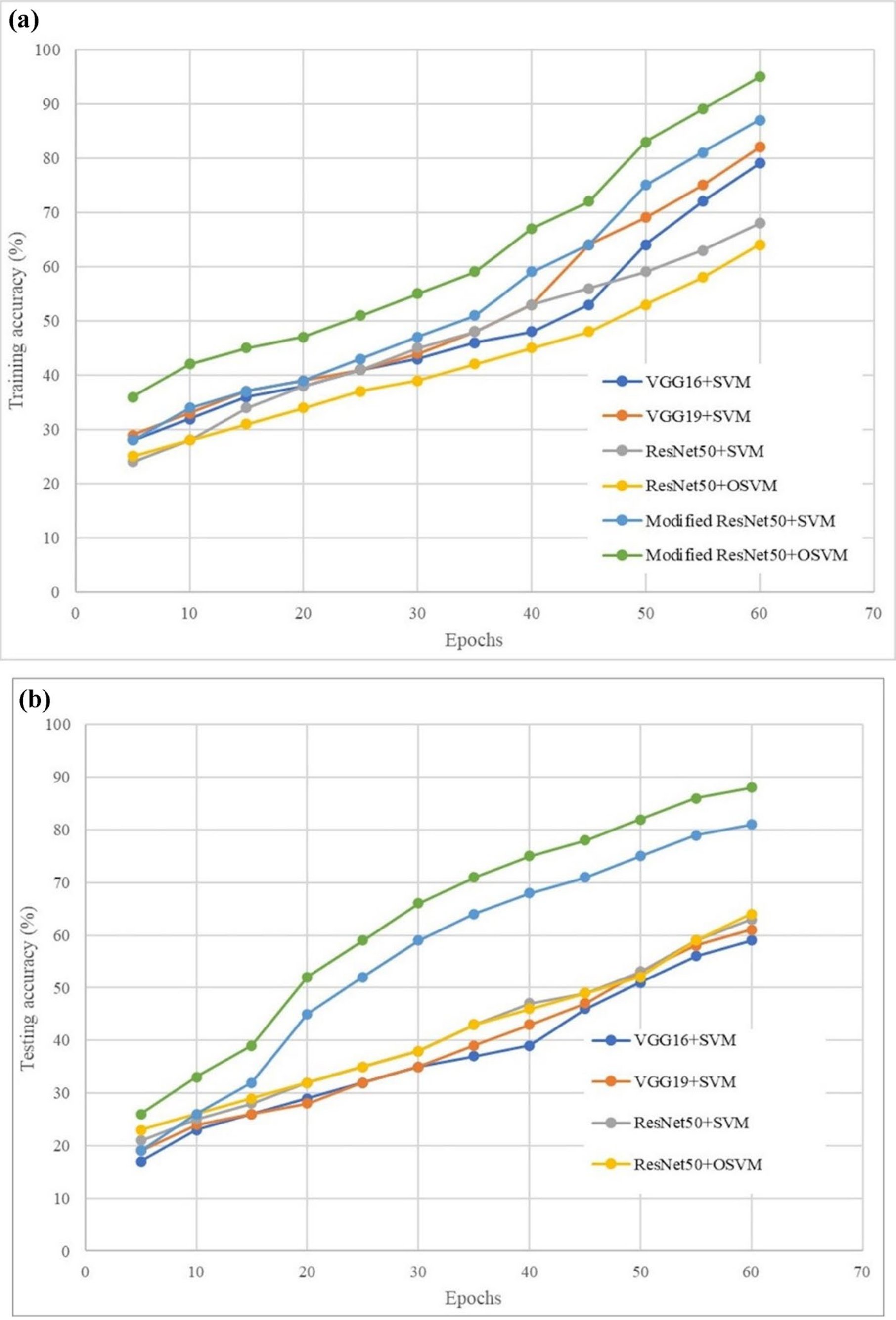


Fig. 3 **a.** Training accuracy analysis of different models. Figure 3 **b**. Testing accuracy analysis of different models

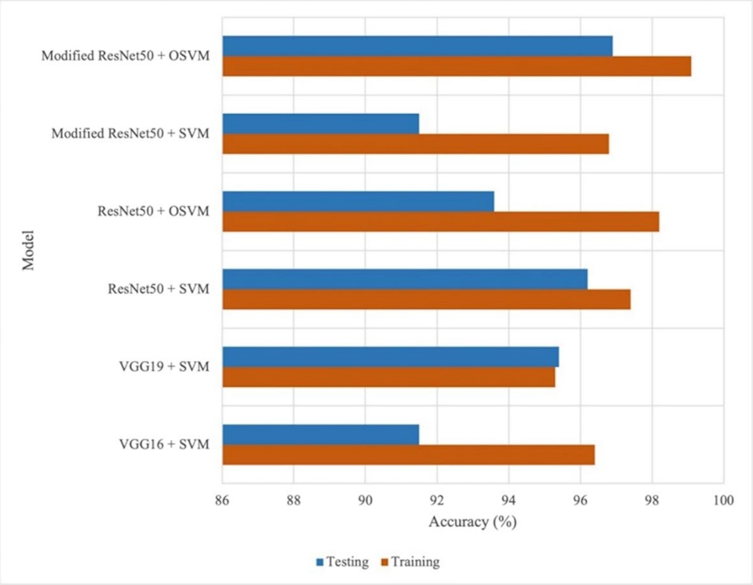


Fig. 4 Simulation average accuracy analysis of the different models

The effectiveness of each of the six distinct models in both training and testing is analysed over a variety of periods. After calculating the mean results of each of the epoch sizes, a measurement is taken, and Eqs. (11) and (12) are used to determine simulation characteristics such as precision and recall.

$$P = \frac{T_{(p)}}{T_{(p)} + F_{(p)}} \tag{11}$$

$$R = \frac{T_{(p)}}{T_{(p)} + T_{(n)}} \tag{12}$$

The values for the true positive, the true negative, and the false positive are represented respectively as $T_{(p)}$, $T_{(n)}$, and $F_{(p)}$. Figure 5a and b show the average training and testing iterations for the six models, respectively. The results show that the updated ResNet50 + OSVM model works well in both training and testing models. In comparison to the other models, this one has a recall percentage that is 14.9% higher and a precision that is 15.3% higher.

The findings of the classification efficiency study of the six distinct models are analysed for both the training stage and the testing stage, and the average results of all the epochs are measured and presented in Fig. 6. Calculating the effectiveness requires the use of Eq. (13).

$$E = \frac{T_{(p)}}{T_{(p)} + T_{(n)} + F_{(p)} + F_{(n)}} \tag{13}$$

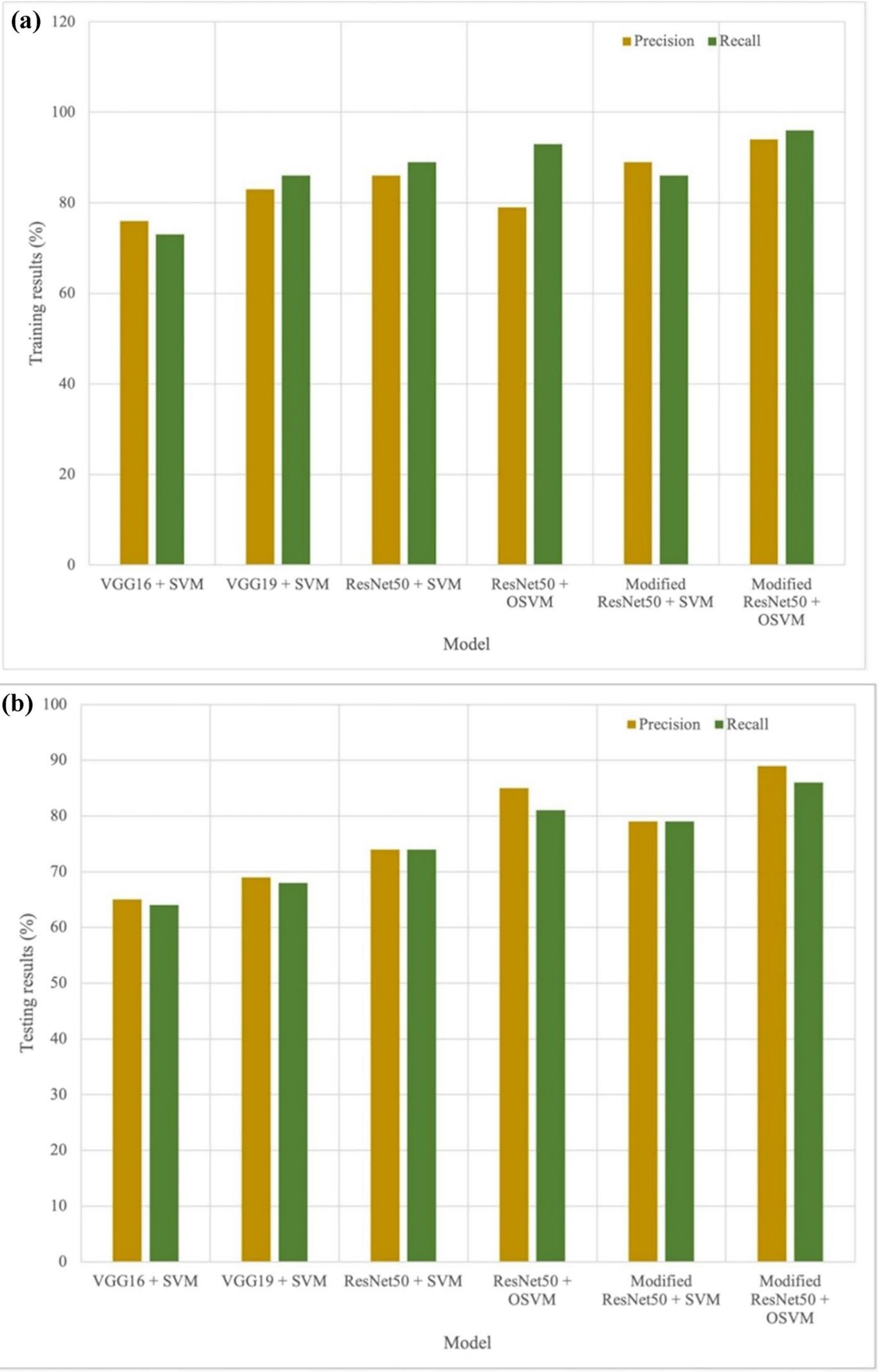


Fig. 5 **a**. Training performance evaluation of the six different models. Figure 5 **b**. Testing performance evaluation of the six different models

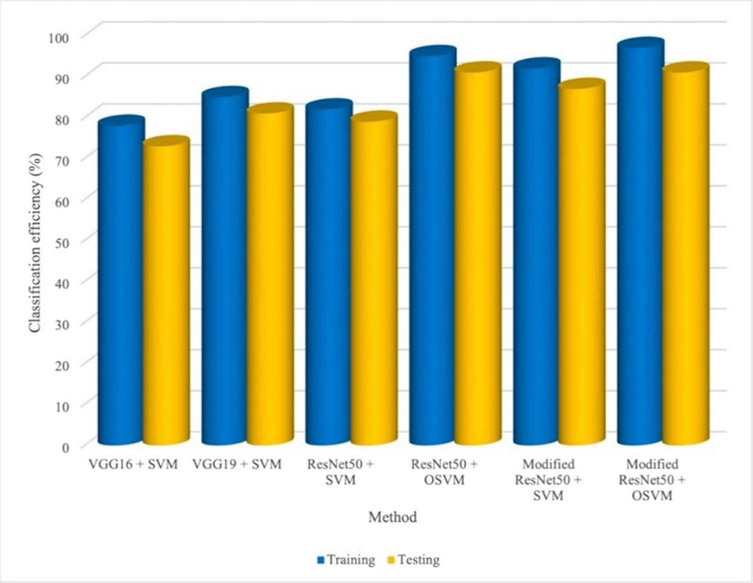


Fig. 6 Classification efficiency analysis of the six different models

The true positive and true negative values are denoted by the notations $T_{(p)}$ *and* $T_{(n)}$respectively, while the false positive and false negative values are denoted by the notations $F_{(p)}$ *and* $F_{(n)}$, respectively. According to the findings, the VGG16 + SVM model performs the worst, while the modified ResNet50 + OSVM model performs the best. The simulation results were discovered using the IMPLAD dataset, and the findings demonstrate that the improved ResNet50 + OSVM model is 18.1% more efficient than the VGG16 + SVM model.

The error analysis of the six distinct models is analyzed, and Fig. 7 depicts the plot of the cumulative results of both training and testing results. Equation (14) is used to calculate the size of the error.

$$e = \frac{T_{(n)} + F_{(p)}}{T_{(p)} + T_{(n)} + F_{(p)} + F_{(n)}} \tag{14}$$

The true positive and true negative values are denoted by the notations $T_{(p)}$ *and* $T_{(n)}$, respectively, while the false positive and false negative values are denoted by the notations $F_{(p)}$ *and* $F_{(n)}$, respectively. According to the findings of the IMPLAD dataset, the modified ResNet50 + OSVM model demonstrates the best performance in terms of the lowest error for both the training (7.2%) and testing (5.7%) phases, whereas the ResNet50 + SVM model demonstrates the worst performance with the highest error for both the training (18.2%) and testing (16.4%) phases. Because of its improved structure and hybrid design, the modified ResNet50 + OSVM model has the lowest error rate.

Analyses are performed on the amount of time needed to evaluate various models, and the findings are displayed in Table 4. When calculating the evaluation time, the whole amount of time needed to finish the entire process, from preprocessing to classification,

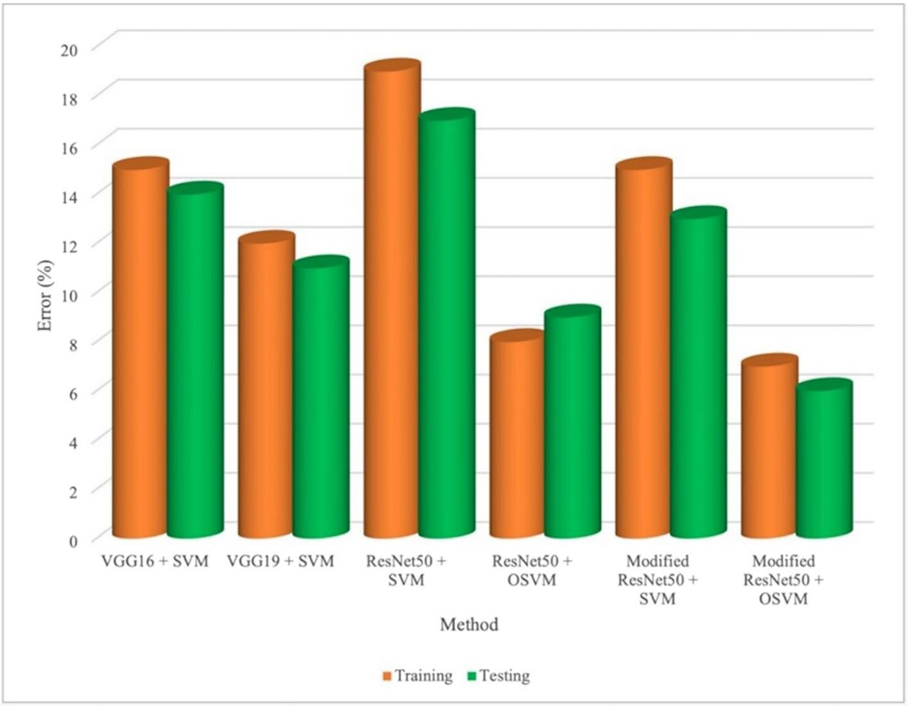


Fig. 7 Error analysis of the six different models

is taken into account. The ResNet50 + OSVM model has a longer training evaluation time than the modified ResNet50 + OSVM model, which is 13,532 seconds, while the modified ResNet50 + OSVM model has the shortest training evaluation time, which is 6853 seconds. Because of its intricate architecture and structure, ResNet50 + SVM takes the least amount of time to complete the testing phase, while ResNet50 + OSVM takes the most.

The simulation results of several models are compared and contrasted in terms of the classification accuracy and error of the IMPLAD dataset for the various leaf pictures. Table 5 displays the outcomes of training and testing for those two parameters, which were measured and displayed in the table. The findings demonstrate that the Modified ResNet50 + OSVM model is superior to the other models in terms of classification efficiency, with an improvement of 24.3% in the training phase, 19.5% in the testing phase, and an error reduction of 51.2% in the training phase and 58.4% in the testing phase,

Table 4 Evaluation time evaluation of different models

Model	Training (sec)	Testing (sec)
VGG16 + SVM	12,324	38
VGG19 + SVM	10,436	34.6
ResNet50 + SVM	11,325	34.3
ResNet50 + OSVM	13,532	63.5
Modified ResNet50 + SVM	1324	54.7
Modified ResNet50 + OSVM	6853	46.5

Table 5 Simulation results analysis of the different models

Model	Classification efficiency (%)		Error (%)	
	Training	Testing	Training	Testing
VGG16 + SVM	78	73	15	14
VGG19 + SVM	85	81	12	11
ResNet50 + SVM	82	79	19	17
ResNet50 + OSVM	95	91	18	19
Modified ResNet50 + SVM	92	87	15	13
Modified ResNet50 + OSVM	97	91	7	6

respectively. When training performance is compared to testing, the training performance is 6.3% better than the testing performance.

The training error and testing error analyses of the six distinct models are analyzed, and the results are assessed by adjusting the epoch size from a minimum of 5 to a maximum of 60 with a step size of 5. The findings show that the epoch size has a significant impact on the accuracy of the models. The results of the training and testing evaluations of errors in the various models are shown graphically in Fig. 8 a and b, respectively, by using Eq. (14). The results of the simulation demonstrate that the error is reduced as the epoch size increases. This is because the larger the epoch, the more times the same leaf picture is processed. According to the findings, the results demonstrate that the Modified ResNet50 with OSVM model performs better than the other models.

4.1 Discussion

The proposed ResNet-50 architecture has been enhanced by effectively max-pooling the activation from the previous fully connected layer to the subsequent convolution layer. In this approach, both the maximum and average activations from the previous convolution are retained, providing the model with knowledge from both perspectives and ultimately improving its performance [32].

To compile the list of online medicinal plant species, the researchers utilized the Indian Medicinal Plants Database (IMPLAD). This comprehensive database served as a valuable resource for gathering information on medicinal plants, facilitating the study's classification objectives [16].

In terms of model comparison, the modified ResNet50 model with an OSVM classifier in the ECNN-PTL approach was evaluated against baseline models such as VGG16, VGG19, and ResNet50. The evaluation considered metrics such as accuracy, precision, recall, error rate, and execution time [32]. The results demonstrate that the modified ResNet50 + OSVM model outperforms the baseline models, achieving a testing phase accuracy of 96.8% and a training phase accuracy of 98.5% [32].

These findings highlight the effectiveness of the proposed model in accurately classifying medicinal plant species. Overall, the enhanced ResNet-50 architecture combined with the OSVM classifier within the ECNN-PTL approach showcases improved performance compared to baseline models. The proposed model demonstrates high accuracy and

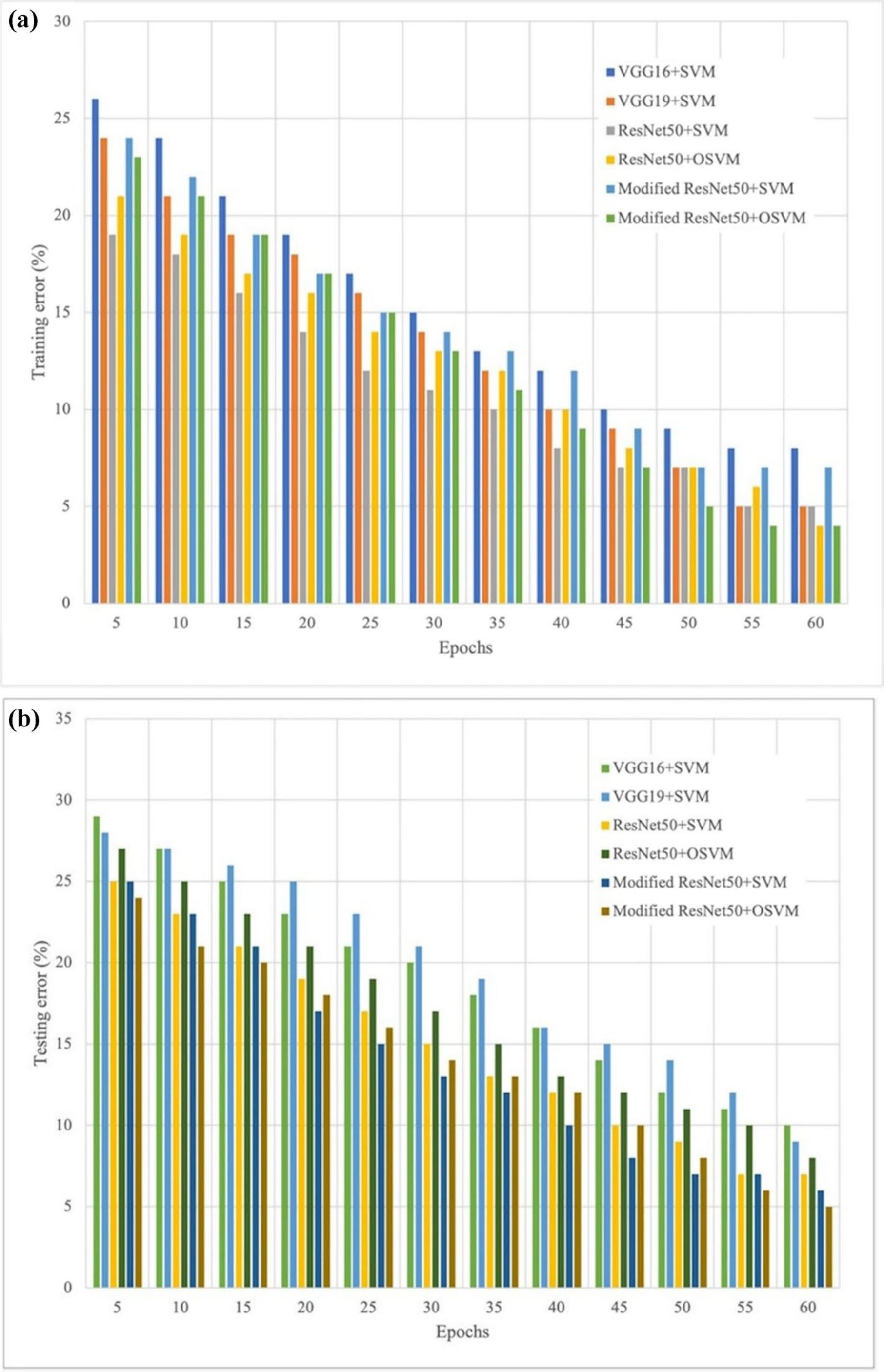


Fig. 8 **a**. Training error analysis of the six different models, Fig. 8 **b**. Testing error analysis of the six different models

Table 6 Comparison of proposed study results with existing methods

Method	Accuracy	Merits	Demerits
MostajerKheirkhah et al. [17]	93.75%	Utilized Probabilistic Neural Network (PNN)	Susceptibility to overfitting, especially when dealing with datasets that have high dimensionality or noisy data
Naeem et al. [18]	93.9%	Texture information extraction from leaf photos	Challenge of accurately capturing and representing complex and fine-grained textures, which may lead to potential loss of important texture details in the process
Kurmi et al. [19]	92.01%	Combined wavelet and temporal dimensions	Increased computational complexity and processing time required for analyzing and extracting features from the data
Pushpa et al. [20]	94.8%	Geometric features and Random Forest (RF) classifier	Limited ability to capture intricate relationships and non-linear patterns in the data compared to more advanced machine learning algorithms
Kaur [21]	95.7%	Geometry and color features with classifiers	Limited capability to capture more complex and abstract patterns in the data compared to more advanced feature extraction techniques
Geetharamani & Pandian [24]	95.7%	9-level deep-CNN for plant pathogen identification	Potential risk of overfitting due to the increased depth and complexity of the network architecture.
Bodhwani et al. [25]	98.7%	50-level recurrent neural network on LeafSnap database	The increased risk of computational complexity and longer training times due to the extensive depth of the network.
Wei Tan et al. [27]	95.1%	D-Leaf CNN model with refined Alexnet	Susceptibility to overfitting due to the increased complexity and number of parameters in the model
Kanda et al. [28]	91.2% (LeafSnap), 99.58% (Flavia)	Logistic regression classification with MobileNet	One potential demerit of using logistic regression classification with MobileNet is its limited capability to capture complex non-linear relationships compared to more advanced classification algorithms

Table 6 (continued)

Method	Accuracy	Merits	Demerits
Sachar et al. [30]	100% (Swedish), 99% (Flavia), 92.4% (MalayaKew)	Feature extraction comparison with VGG-16, Xception, MobileNetV2, DenseNet121	High resource and computation requirements
Proposed Study	**96.8%**	High accuracy, improved ResNet50 architecture	Higher computational requirements compared to simpler models

robustness in classifying medicinal plants, contributing to the field of automated medicinal plant identification and classification [32].

When comparing the model consisting of ResNet50 and an SVM classification that was fine-tuned using the optimization approach, the findings obtained demonstrate that ResNet50 in conjunction with OSVM displays high-accuracy results. In addition, the presented IMPLAD model may be used for any medicinal plant categorization issues due to the proposed method's accuracy in less execution time. Additionally, the research demonstrates that PTL is an effective method for constructing any CNN model despite having a constrained dataset. Table 6 shows the comparison of proposed study results with existing methods.

5 Conclusion, limitation and scope for future work

An Enhanced Convolutional Neural Network architecture (using modified ResNet50) with Progressive Transfer Learning (ECNN-PTL) has been proposed in this paper. The suggested method uses an improved ReNet50 framework for feature extraction along with PTL. PTL has been employed in the two learning steps, and the image size is allowed to gradually rise from 64, 128, and 150 to 256 pixels. Classification has been done using an Optimized Support Vector Machine (OSVM). The classical SVM hyperparameters are tuned further by the Adam optimizer to achieve a better performance model. Image preprocessing, segmentation, extraction of features, and classification are the four stages of the proposed work. The first step is to capture digital pictures of the plant specimens. The segmentation and pre-processing phases receive the leaf pictures from the medicinal plant for further extraction of features and classification. Using the updated ResNet50 architecture and PTL, the leaf's attributes are extracted to acquire significant data. Finally, the attributes are categorized using SVM and OSVM algorithms.

The modified ResNet50 + OSVM model performs better than the previous models in both the training and testing stages, according to the overall simulation results. The modified ResNet50 + OSVM model had a testing phase accuracy of 96.8% and a training phase accuracy of 98.5%. Owing to the proposed techniques' accuracy and speedy execution, the presented IMPLAD dataset may also be employed for the classification of medicinal plants. Additionally, the study shows that PTL, even with a limited dataset, is a successful strategy for building any CNN model. The IMPLAD model also has the drawback of being unable to differentiate between compound leaves and leaf images placed on complex backgrounds.

Future studies will concentrate on expanding IMPLAD dataset to mobile applications, giving everyone quick access to the knowledge of medicinal plants. Additionally, enhancing the classification method to recognize extremely similar groups will enable strengthening the accuracy of the suggested system. The field of medicinal herbs requires immediate action to find and gather endangered plant species from remote locations. Many plants can be replanted, which can help the nation's biodiversity while also providing a natural cure for a variety of illnesses.

Data availability Data sharing is not applivable to this auricle as no datasets were generated or analyzed during the current study.

Declarations

Conflict of interest Dear editor.
I have no conflict of interest.

References

1. Austen GE, Bindemann M, Griffiths RA, Roberts DL (2016) Species identification by experts and non-experts: comparing images from field guides. Sci Rep 6(1):1–7
2. Rull V (2022) Biodiversity crisis or sixth mass extinction? Does the current anthropogenic biodiversity crisis really qualify as a mass extinction? EMBO Rep 23(1):e54193
3. Van Horn G, Mac Aodha O, Song Y, Cui Y, Sun C, Shepard A. ... Belongie S (2018) The inaturalist species classification and detection dataset. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 8769–8778
4. Willis CG, Ellwood ER, Primack RB, Davis CC, Pearson KD, Gallinat AS, … Soltis PS (2017) Old plants, new tricks: Phenological research using herbarium specimens. Trends Ecol Evol 32(7):531–546
5. Zhang C, Lu Y (2021) Study on artificial intelligence: the state of the art and future prospects. J Ind Inf Integr 23:100224
6. Wang A, Zhang W, Wei X (2019) A review on weed detection using ground-based machine vision and image processing techniques. Comput Electron Agric 158:226–240
7. Yang C (2021) Plant leaf recognition by integrating shape and texture features. Pattern Recogn 112:107809
8. Li Z, Guo R, Li M, Chen Y, Li G (2020) A review of computer vision technologies for plant phenotyping. Comput Electron Agric 176:105672
9. Mohtashamian M, Karimian M, Moola F, Kavousi K (2021) Masoudi-Nejad, A (2021) Automated plant species identification using leaf shape-based classification techniques: a case study on Iranian maples. Iran J Sci Technol Trans Electr Eng 45:1051–1061. https://doi.org/10.1007/s40998-020-00398-2
10. Armi L, Fekri-Ershad S (2019) Texture image analysis and texture classification methods-A review. arXiv preprint arXiv:1904.06554
11. Larese MG, Namías R, Craviotto RM, Arango MR, Gallo C, Granitto PM (2014) Automatic classification of legumes using leaf vein image features. Pattern Recogn 47(1):158–168
12. Goyal N, Kumar N (2022) Leaf bagging: a novel meta heuristic optimization based framework for leaf identification. Multimed Tools Appl 81(22):32243–32264
13. Shaheen S, Ramzan S, Khan F, Ahmad M, Shaheen S, Ramzan S, ... Ahmad M (2019) History, classification, worldwide distribution and significance of herbal plants adulteration in herbal drugs: a burning issue, pp 35–49
14. Máthé Á, Khan IA (2022) Introduction to medicinal and aromatic plants in India. In: Medicinal and aromatic plants of India, vol 1. Springer International Publishing, Cham, pp 1–34
15. Roopashree S, Anitha J (2021) DeepHerb: a vision based system for medicinal plants using xception features. IEEE Access 9:135927–135941
16. Nazarenko DV, Kharyuk PV, Oseledets IV, Rodin IA, Shpigun OA (2016) Machine learning for LC–MS medicinal plants identification. Chemom Intell Lab Syst 156:174–180
17. MostajerKheirkhah F, Asghari H (2019) Plant leaf classification using GIST texture features. IET Comput Vis 13(4):369–375
18. Naeem S, Ali A, Chesneau C, Tahir MH, Jamal F, Sherwani RAK, Ul Hassan M (2021) The classification of medicinal plant leaves based on multispectral and texture feature using machine learning approach. Agronomy 11(2):263
19. Kurmi Y, Gangwar S, Chaurasia V, Goel A (2022) Leaf images classification for the crops diseases detection. Multimed Tools Appl 81(6):8155–8178
20. Pushpa BR, Lakshmi P (2022) Deep learning model for plant species classification using leaf vein features. In: 2022 international conference on augmented intelligence and sustainable systems (ICAISS). IEEE, pp 238–243
21. Kaur S, Kaur P (2019) Plant species identification based on plant leaf using computer vision and machine learning techniques. J Multimed Inf Syst 6(2):49–60
22. Turkoglu M, Hanbay D (2019) Recognition of plant leaves: an approach with hybrid features produced by dividing leaf images into two and four parts. Appl Math Comput 352:1–14
23. Hu J, Chen Z, Yang M, Zhang R, Cui Y (2018) A multiscale fusion convolutional neural network for plant leaf recognition. IEEE Signal Process Lett 25(6):853–857
24. Geetharamani G, Pandian A (2019) Identification of plant leaf diseases using a nine-layer deep convolutional neural network. Comput Electr Eng 76:323–338
25. Bodhwani V, Acharjya DP, Bodhwani U (2019) Deep residual networks for plant identification. Proced Comput Sci 152:186–194
26. Cheplygina V, de Bruijne M, Pluim JP (2019) Not-so-supervised: a survey of semi-supervised, multi-instance, and transfer learning in medical image analysis. Med Image Anal 54:280–296

27. Wei Tan J, Chang SW, Abdul-Kareem S, Yap HJ, Yong KT (2018) Deep learning for plant species classification using leaf vein morphometric. IEEE/ACM Trans Comput Biol Bioinf 17(1):82–90
28. Kanda PS, Xia K, Sanusi OH (2021) A deep learning-based recognition technique for plant leaf classification. IEEE Access 9:162590–162613
29. Taheri-Garavand A, Nasiri A, Fanourakis D, Fatahi S, Omid M, Nikoloudakis N (2021) Automated in situ seed variety identification via deep learning: a case study in chickpea. Plants 10(7):1406
30. Sachar S, Kumar A (2021) Automatic plant identification using transfer learning. In IOP conference series: materials science and engineering (vol 1022, no. 1). IOP Publishing, p 012086
31. Kaya Y, Ercan GÜrsoy. (2023) A novel multi-head CNN design to identify plant diseases using the fusion of RGB images. Ecol Inf 75:101998
32. Keceli AS, Kaya A, Catal C, Tekinerdogan B (2022) Deep learning-based multi-task prediction system for plant disease and species detection. Ecol Inf 69:101679
33. Chen D, Yuzhen L, Li Z, Young S (2022) Performance evaluation of deep transfer learning on multi-class identification of common weed species in cotton production systems. Comput Electron Agric 198:107091
34. Huang M-L, Chuang T-C, Liao Y-C (2022) Application of transfer learning and image augmentation technology for tomato pest identification. Sustain Comput Inf Syst 33:100646
35. Venugopalan Nair SN, Ved DK, Ravikumar K, Tabassum IF, Sureshchandra ST, Somasekhar BS, ... Shankar D (2020) Indian medicinal plants database (IMPLAD) and threatened medicinal plants of India. Conservation and Utilization of Threatened Medicinal Plants, pp 63–92

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.