Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 235 (2024) 1135–1144

Procedia Computer Science

www.elsevier.com/locate/procedia

International Conference on Machine Learning and Data Engineering (ICMLDE 2023)

Ethno medicine of Indigenous Communities: Tamil Traditional Medicinal Plants Leaf detection using Deep Learning Models

G.B.Govindaprabhu[a], M.Sumathi[b]

[a]Research Scholar (MKU23PFOS10909), Madurai Kamaraj University, Madurai. orcid: 0000-0002-2297-3597

[b]Head & Associate Professor, PG & Research Department of Computer Science, Sri Meenakshi Government Arts College for Women, Madurai.

Abstract

In indigenous communities, ethno medicine has played a prominent role in healing for centuries and provides valuable insight into the use of traditional medicinal plants. Using local plant leaves as therapeutic agents, this research explores the importance of traditional Tamil medicine. Traditional healers and practitioners can benefit from technological advances and collaborate with modern healthcare systems through these advances. Based on deep learning models, a novel approach for recognizing leaves of traditional Tamil medicinal plants is presented. Pharmaceutical companies are increasingly paying attention to plants that contain medicinal factors that have fewer side effects during treatment. By identifying plants with medicinal factors and identifying their medicinal uses, this work addresses the identification of plants with medicinal factors. To classify medicinal plants and their uses, we will use deep learning models and image processing techniques to identify Tamil Traditional Medical Plant (TTMP). The input dataset for this work includes 18 different traditional Tamil plant leaves such as Azadirachta Indica (Neem), Ocimum Tenuiflorum (Tulsi) and Trigonella Foenum-graecum (Fenugreek). Pre-processing steps are performed on the input plant image, including RBZR Augmentation, noise removal and grayscale conversion, to detect the image area that is important for plant type identification. To identify the core area of the plant, this work using the HGAW Segmentation algorithm. Deep learning model is trained with segmented plants and their labels as inputs. The aim is to digitize and make accessible ancestral medical knowledge by incorporating deep learning-based leaf recognition with 96.71% accuracy for the proposed EEXR model.

Keywords: Type your keywords here, separated by semicolons ;

1. Introduction

For centuries, ethno medicine has played a prominent role in healing and provides valuable insight into the use of traditional medicinal plants in indigenous communities [1]. Using local plant leaves as therapeutic agents, this research explores the importance of traditional Tamil medicine. The geographical distribution of these medicinal

10.1016/j.procs.2024.04.108

plants has changed dramatically due to the effects of climate change [2], making their identification difficult. They are rare to find, as most have become endangered species due to habitat loss [3]. A traditional Tamil Siddha doctor in ancient times selected medicinal plants from the fields and prepared them for his patients. The production and marketing of Siddha medicines has developed into an important industry in recent decades. Identifying plants with medicinal properties and their uses is crucial as studies have shown that certain plants contain compounds that can treat diseases [4]. However, nowadays common people has lack of awareness about such medicinal plants. Plants like these Siddha medicinal plants are different from normal plants. As early as 200 years ago, when people recognized their healing properties, various diseases were cured with these plants. Typically, these plants were found in backyards or along roadsides. Identifying medicinal plants has become increasingly difficult in recent years. Most people lack awareness about these plants. Identifying the leaves of plants is the first step toward classification. [5].

Additionally, identifying plants with medicinal properties is helpful for some people because they can use them daily. A key aspect of the medical industry is the identification of plant leaves. Developing innovative medical ideas is of utmost importance. Viruses and diseases are spreading at an unprecedented rate, requiring new treatments. Identification of plant leaves is crucial to medicine as new viruses such as COVID-19 emerge [6] and new treatments are essential. Pharmaceutical companies are researching plants with medicinal properties that have fewer side effects [7]. By digitizing ancestral medical knowledge and enabling collaboration between traditional and modern medicine, technological advances in medicinal plant identification can benefit practitioners [8]. This work deals with medicinal plant identification, classification using image processing and deep learning techniques.

i) Key Contributions:

- Proposes a novel approach for detecting and classifying medicinal plant leaves based on image processing and deep learning techniques. Introducing a hybrid genetic algorithm with watershed method (HGAW) for accurate segmentation of leaf images.
- Develops an ensemble model for deep neural networks that incorporates the EfficientNet,
- Provides a way to digitize and preserve traditional ethno medical knowledge using AI, which can benefit practitioners.

ii) Paper Structure:

Section II, Literature Review, summarizes related work on leaf classification techniques. Section III - Proposed model explaining the overall methodology and detailed algorithms for pre-processing, segmentation, feature extraction and classification. Section IV, Performance Evaluation, analyses the results of the proposed model using appropriate metrics such as accuracy, precision and recall. Section 5 provides a conclusion summarizing the contributions of the work and the effectiveness of the proposed approach.

2. Review of Literature

Aitwadkar et.al [9] proposed a simple and effective method for identifying medicinal plants using descriptors such as edge, area and color extracted from leaves. Using machine vision and digital image processing, Pushpa et.al [10] explains how Ayurvedic plants can be classified. In this methodology, three phases are followed: preprocessing, feature extraction, and classification. With the proposed methodology, an accuracy rate of 93.75 percent is achieved. An automated system for identifying and describing medicinal plants is created using a vision-based approach as proposed by Venkataraman and his team [11]. The purpose of this work is to discuss how the feature set is formed, which is the key to recognizing any plant species.

According to Janani [12] proposed the idea to extracted, shapes, colors and textures from leaf images and used in conjunction with artificial neural networks (ANNs) to identify leaf classes. To achieve high efficiency and minimal complexity, selecting the right image input functions is crucial. The accuracy of the network was tested using various combinations of features. This method was tested on 63 leaf images with eight input features and achieved an accuracy of 94.4%. As a means of classifying plant leaves, Meeta et.al [13] presented an overview of various classification techniques. Patterns are classified according to their inputs into the classification problem. Identifying leaves based on their morphological characteristics is an example of plant leaf classification. To maximize the identification rate, Manojkumar's work examines both front and back feature vectors as well as morphological features [14]. Combinations of unique characteristics are used to classify leaves. An identification rate of 99% was achieved using various classifiers.

According to Wu [15], probabilistic neural networks and image processing techniques have been used to automate leaf detection for plant classification. Using 12 leaf features, five main variables are orthogonally transformed into a PNN input vector. By analyzing 1800 leaves, it is possible to classify, 32 plant species with an accuracy of 95%. This work [16] implemented a leaf detection system based on SVMs to extract and classify RSC features in leaf images. Biological, forestry and agricultural studies have found that automatic identification and classification of plants in botanical gardens are useful tools for the study and discovery of new species. By identifying medicinal plants, herbal medicines can also be produced.

Keskar et.al [17] identifies the therapeutic plants commonly used in Ayurveda by automating the process. Medicinal plants can be distinguished by their leaves' shape, color, and texture. A hybrid optimal machine-learning algorithm is presented for ayurvedic medicinal plant identification (EAC-AMP). A number of methods have been proposed by Gokhale et.al proposed the system [18] for identification of medicinal leaves, but their effectiveness varies. Plants can be identified from their leaves using CNNs in this study. In order to train the model, this method created a CNN model from scratch and used Tensorflow Object Detection API to detect these plants.

3. Proposed Model

A proposed model consists of four phases is shown in Fig 1. Preprocessing, region detection, feature extraction, and identification. By subtracting the background from an image, region detection finds its structure in the preprocessed image. Machine learning techniques are used to extract features from the images and then three algorithms are used to identify the categories. This framework performs image enhancement and noise reduction as part of the preprocessing phase. Another identification process is carried out by detecting external boundary areas for segmentation. Plant images are classified based on their color and shape.

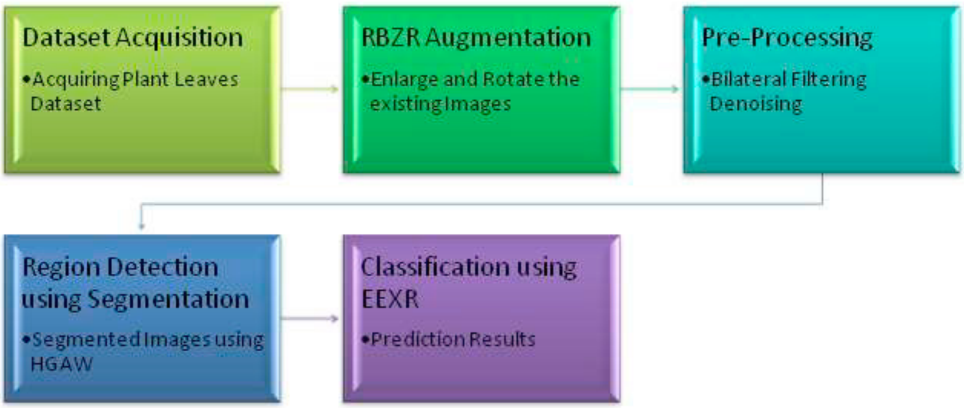


Fig.1. Proposed Model Process Flow

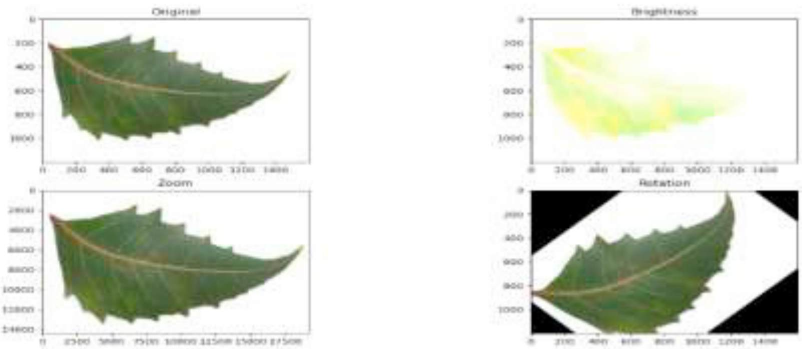
3.1 Dataset

Based on the Mendeley dataset [19], it contains a variety of plant species. There are 30 species in this dataset, including Santalum album (sandalwood), Muntingia calabura (Jamaica cherry), and many others. The dataset includes 1800 images representing 18 species shown in Fig 2. The number of high-quality images for each species is between 80 and 100. A single folder is named after the scientific/botanical name of the species.

Fig. 2. Plant Leave Dataset.

3.2 RBZR Augmentation

The process of expanding images involves developing more data for model training by modifying existing data. Deep learning models are therefore trained using artificially expanded datasets. In RBZR models, the expansion methods include random brightness, zoom, and rotation. Rotating an image is the most common technique for enlarging it. There is no difference in the information regardless of which way the image is rotated. In Fig 3, different rotation angles and brightnesses are shown. Different angles make the data appear different, regardless of the fact that the input data has not changed.



.

Fig 3. RBZR Images

3.3 Pre-Processing

The processing of digital images involves the application of computer algorithms. As part of image processing, it is imperative to remove image noise (de-noise). Images are inevitably contaminated with noise during the capture, compression and transmission process due to the environment, transmission channel, etc., resulting in distortion and loss of information. It is the process of removing noise from an image to make it more visible. Noise removal is about recovering meaningful information from noisy images to obtain high-quality images.

3.3.1 Bilateral Filter

The bilateral filter can be used to blur a sharp image while keeping edges sharp using a linear Gaussian filtering technique. Because it allows a photo to be split into different scales without creating a halo, it is often used for applications such as tone mapping, style transfer, relighting, and noise reduction.

$$g(X) = (f * G^S)(X) = \int_R f(Y) G^S(X-y)dy$$

The weight for $f(Y)$is the same $G^S(X-y)$and depends only on the spatial distance ‖x−y‖. The bilateral filter adds a weighting term that depends on the tonal distance $f(Y) - f(x)$ this result in:

$$g(X) = \frac{\int Rf(y)Gs(x-y)Gt(f(x)-f(y))dy}{\int_R G^S(X-y)G^t(f(X)-f(Y))dy}$$

Algorithm 1: Bilateral Filter
Input: Img_{In} – Input Image, **Output:** Img_D – Denoised Image
1. Read Image from Img_{In}
2. Compute geometric distance from all each other pixels (x,y)
3. d(x',y') = √(x - x')^2 + (y - y')^2 // d is the distance
4. Compute the photometric distance for each pixels (x,y) and (x', y')
5. Calculate the filters weights for distance&Normalize the filter weights
6. Calculate the filtered output as pixel
7. Img_D = ∑ w(x',y') * I(x',y') // w is weights for distance
8. Return Img_D

End Algorithm

Algorithm 1. Bilateral Filter Denoising

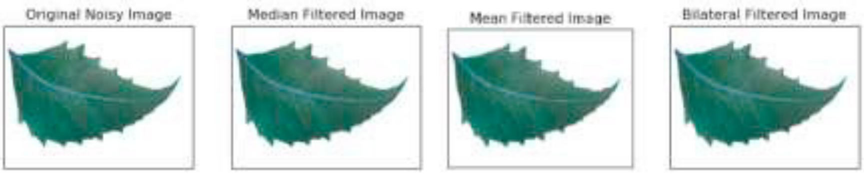


Fig 4. Bilateral Filtered Image

Fig 4 describes the Bilateral Filtered Image. The figure illustrates the denoised images using median filtered, mean filtered and bilateral filtered images.

3.4 Region Detection with Segmentation

This section contains the region detection (Segmentation) HGAW algorithm for the plant leaf identification.

3.4.3 Hybrid Genetic Algorithm With Watershed (HGAW)

This segmentation is hybrid with watershed and genetic algorithm. Genetic algorithm is an adaptive method that can solve optimization and search problems. Organisms function based on their genetic makeup. There are two steps during this process. In the first step, the crossing points and parents are selected uniformly. Two new individuals are then created by exchanging alternative choice pairs between the two selected locations. Gene values can be randomly changed through mutation and evolve until they reach optimization. This makes GA capable of global optimization and can avoid crashing into local optima despite the loss of some chromosome information during selection. To tidy up the image, the Watershed segmentation algorithm connects a set of pixels within the foreground objects. The "impose_min" function can be used to select a specific location as the range of minima. Each object in the label matrix is assigned a color by combining a color map with a set of objects in the matrix.

Algorithm 2: HGAW
Input: Img_D – Denoised Image, **Output:** Img_{Seg} – Segmented Image
1. Read image from $Img_{D\&}$Convert_Grayyscale (Img_D)
2. M=Find (Peak_local_Max (p, Img_D))
3. Apply watershed segmentation
4. Img_{Seg} = Morphology_watersheld(Img_D, M)&Computer threshold 't' to remove small regions
5. If (Img_{Seg}< t)
   a. Refine GA by population P with size of N
   b. Calculate fitness F(p) = sum(Img_{Seg} * p) / sum(Img_{Seg})
   c. Perform crossover and mutation on parents to create offspring O
   d. Replace least fit individuals in P with individuals in O
6. Return Img_{Seg}

End Algorithm

Algorithm 2. GA with Watershed Segmentation Algorithm

At beginning, the process initializes the population (pop, size N). In the next step, it evaluates the fitness of each agent. The creation of new populations is carried out through GA selection, crossover and mutation operations in the third step (GA method). The fourth step involves updating G (t), best(t), worst(t) and $M_i(t)$. The current iteration number is t and currently N iterations have been performed; t represents the number of the current iteration. The total force is calculated over different watershed values. The background image has "holes" that can be filled with flood fill. In step 6, the matrix is reverse transformed by changing zeros to 1 and vice versa. The seventh step involves updating the location of the water catchment areas. Steps 5 to 7 are repeated in Step 8. Based on FCM, Otsu and HGAW segmentation of datasets, Fig 5 shows segmentation results for datasets.

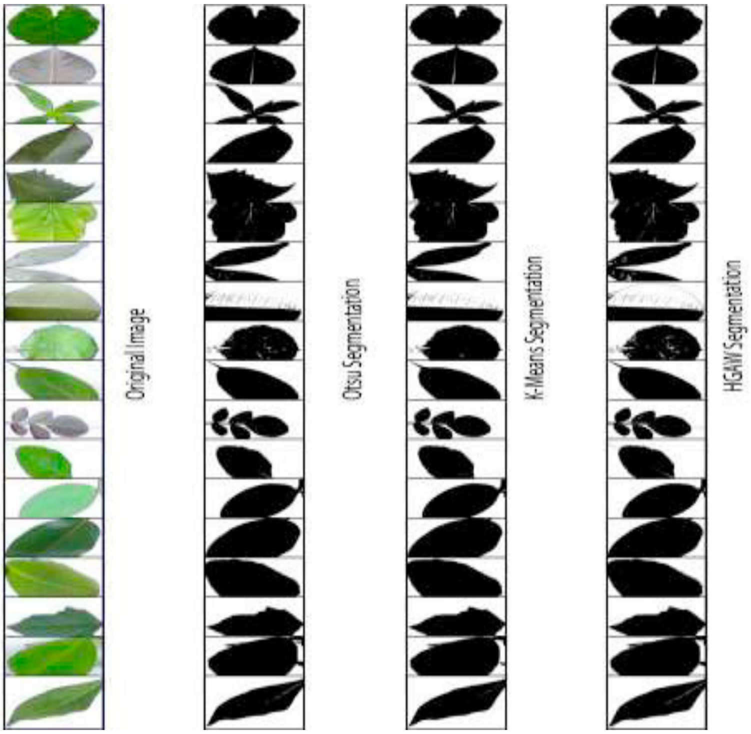


Fig. 5: Segmentation Result

3.5 Classification

To identify the images, the features constructed from the images are used for plant classification. Using this level, the work identifies the leaf pattern of plants and provides suggestions for learning.

3.5.1 EEXR (Ensemble EfficientNet and Xeception with ResNet)

This ensemble model is combine the three algorithms EfficientNet and Xeception with ResNet and the algorithms are explained as follows.

a) EfficientNet

. Using this Efficient Net, the work identifies the leaf pattern of plants and provides suggestions for learning.

$$depth(d) = \alpha^{\emptyset}, width(w) = \beta^{\emptyset}, resolution(r) = \gamma^{\emptyset}$$
$$\boldsymbol{Where\ \alpha \geq 1, \beta \geq 1, \gamma \geq 1,}$$
$$\emptyset = \boldsymbol{determine\ the\ amount\ of\ effective\ resource\ extension\ model}$$

A multi-class image classification experiment was conducted using the EfficientNet classifier and the Tamil Traditional Medicine (TTM) dataset. Each class contains approximately 1800 images, divided into 18 categories. Training and tests were carried out for 18 different classes. This EfficientNet model has five dense layers with input sizes of 1600x1200x3 and 800x600x2. It also has an 800 filtered input layer. Another dense layer: 400 x 300 x 2 kernels that filter the inputs from the previous layer. Filters with a core size of 200 x 150 x 2 are arranged in three dense layers. There are also 100x75x2 kernels in the fourth layer of dense inputs. Also, SOFTMAX activation is applied to the output layers with a batch size of 100 and an epoch size of 18x1. With categorical cross-entropy loss for accuracy validation, the model uses stochastic gradient descent as the optimizer. The batch size is 100, the epochs are 50, and the learning rate is .001.

The Xception module replaces the standard Inception modules with a single module that consolidates depth-separable convolutions. For folds that can be separated in depth, the depth level is followed by the point level. A channel-wise convolution of N x N is equivalent to a depth-wise convolution of N x N. This Xception model has a first dense layer with a size of 800 x 600 x 3. Using the input layer, we filter the images 1600 x 1200 x 3. A Kernel with size 400x300x2 filters inputs of 400 in the dense layer. In the third dense layer, the kernel size is 200. Four and five dense layers contain 200x150x1 inputs and kernels, respectively. As part of the final steps, SOFTMAX activation was used to create epoch layers 18 x 1 with a batch size of 100 for output to the classifier. A categorical metric for cross-entropy loss validation is included as well as Adam optimization. ResNet uses a jump connection (or shortcut connection) to fit input from a previous layer directly into a subsequent layer without changing the input. It enables deeper networking through skip connections. Added a jump/shortcut connection after the weight layers to solve the issue of disappearing gradients. Fig. 6 shows the ResNet weight calculation. The result is below:

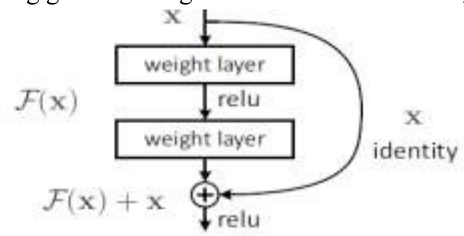


Fig. 6: ResNet weighting Calculation

Therefore, the output is H(x)= F(x) + x. The weight layer is actually used to learn a kind of residual map: F(x)=H(x)-x. Using ResNet, a model with size 1600 x 1200 x 3, an input layer of 1600 with the first dense layer of 1600 x 1200 x 3. The second dense layer consists of kernel sizes of 800 x 600 x 2, the inputs of 1600 filter. A In the third dense layer there is a filter with kernel size 400. Likewise, the kernel is 200 x 150 x 2 with inputs of 400 in the fourth dense layer. Next, the output layers that provide the output to the classifier were prepared with batch size 100 and size 18x1. Ensemble conditions return the value of a if a and b are equal. This function returns the value of b if b and c are equal. As long as a and c are equal, c is returned. The EfficientNet, Xception model and ResNet model are represented by a, b and c, respectively.

Algorithm 3: EEXR Classifier
Input: F_S – Feature Set, **Output:** C_{Label} – Predicted Class

1. Fetch features set F_S, Train EfficientNet, Xception, and ResNet models&Create a test dataset and load it
2. Calculate the predicted probabilities for each class of images in the test dataset by passing each image through all three models
3. Let E(x), X(x) and R(x) be the predicted probabilities for image x for all models
4. EXR(x)=EEXR.Predict (x)&Combine all the predict probabilities
5. f(E(x), X(x), R(x)) = w1 * E(x) + w2 * X(x) + w3 * R(x) // f is ensemble function and w1,w2 & w3 are weights of the model
6. Assign the class with the highest probability as the final prediction for the image x
7. C_{Label} = argmax(P(x)) // P(x) is prediction possibilities
8. Repeat steps for all image features set
9. Return C_{Label}

End Algorithm

Algorithm 3. EEXR Classifier Algorithm

4. Performance Evaluation

a) PSNR

PSNR (Peak Signal-to-Noise Ratio), also called Peak SNR, refers to the ratio between the maximum possible power (of a signal) and the possible distortion of the signal. Unlike other filters, high PSNR values are considered indicators of high-quality processing. Fig. 7 shows that bilateral compared to PSNR values is better than other filters.

b) MSE

The MSE value represents the average difference between pixels in an image. If the MSE value is higher, there is a larger difference between the original image and the preprocessed image. However, edges must be handled very carefully.

c) SSIM

SSIM (Structural Similarity Index) measures image loss caused by processing, such as compression or loss in transmission. Fig 7 shows SSIM values. Fig 7 shows the PSNR, MSE and SSIM values for median, mean and bilateral filters.

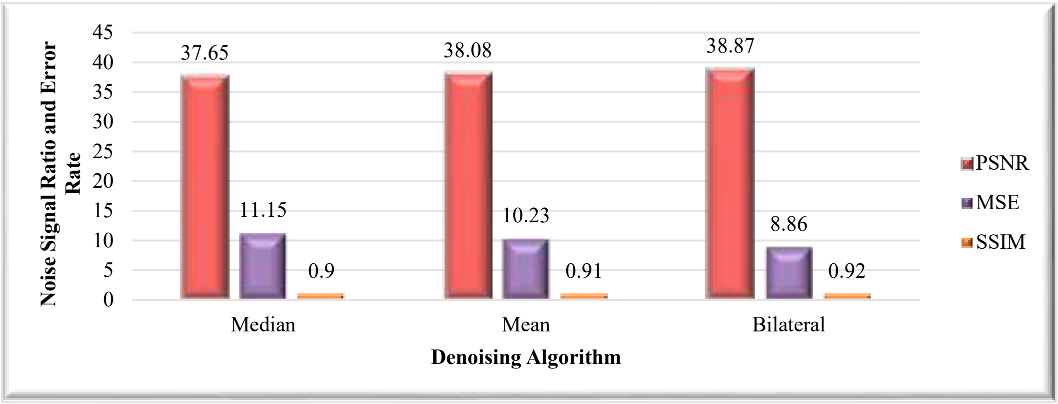


Fig. 7: PSNR, MSE and SSIM

d) Accuracy

Based on the input data in a dataset, the accuracy of a machine learning model measures how well a model can identify relationships and patterns between features. The error rate indicates the proportion of incorrectly classified instances in the total set of instances. Fig 8 shows the accuracy and error rate.

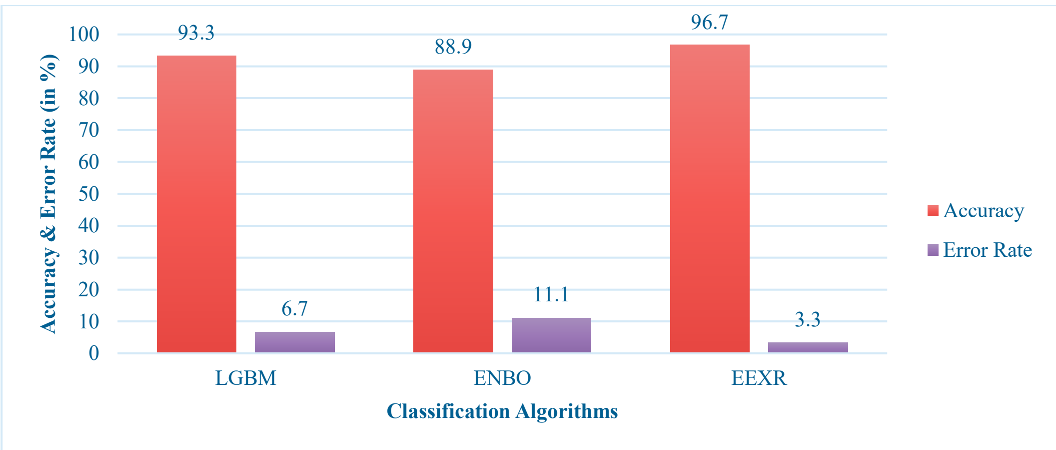


Fig. 8: Accuracy and Error Rate

f) Precision & Recall

The calculation accuracy must take into account the ratio of all positive results to all true positive results. In other words, our recall gives us an idea of how well our model identified true positives. Models may need to allow a certain amount of false negatives to come through. False negatives would not affect the recall equation, resulting in higher precision. There are times when a model would benefit from allowing more false positives through, which would result in higher recall because false positives are not taken into account. Figure 9 shows the precision and recall. F-scores, also called F1-scores, are used to assess how accurate models are. The F-score is calculated by taking the harmonic mean of the model's precision and recall and calculating the F-score. The f1-score of the classifiers is shown in Fig 9.

Fig. 9: Precision- Recall and F1-Score

We compare the performance of our proposed EEXR ensemble model with state-of-the-art medicinal plant classification methods. Keskar et al. [17] developed a model using the optimal WO-DNN architecture and achieved 91.3% accuracy in classifying Ayurvedic medicinal plants. Our approach achieves a significantly higher accuracy of 96.71% on a similar task. Gokhale et al. [18] implemented CNN models such as VGG-16, ResNet50 and Inception-V3 for leaf-based identification of medicinal plants. Their best model, ResNet50, achieved 92.7% accuracy. In comparison, our ensemble technique outperforms the accuracy by about 4%. Manojkumar et al. [14] extracted texture and shape features from plant leaves coupled with SVM and MLP classifiers. Their approach yielded a maximum accuracy of 93.6%, while our deep learning method outperforms this value.

The key factors contributing to the superior performance of our proposed technique include:

- Effective pre-processing with bilateral filters and HGAW segmentation,
- Combining several state-of-the-art deep CNN architectures such as EfficientNet and Xception,
- Introducing residual connections over ResNet that improve gradient flow, rich data
- Extension to expose the model to large variations.

Our proposed medical leaf classification framework outperforms current approaches in the literature by 2-4% in accuracy.

5. Conclusion

This work demonstrates a technique for automated detection and classification of medicinal plant leaves using image processing and deep neural networks. As an application example, the importance of digitization and preservation of traditional ethnobotanical knowledge is highlighted. The proposed methodology follows a systematic three-stage pipeline – preprocessing, segmentation and classification. Standard image enhancement techniques are used to remove noise and unwanted artifacts. A novel Hybrid Genetic Algorithm with Watershed (HGAW) technique is introduced to accurately separate the leaf from its background. Robust feature vectors including shape and colour data are constructed to train various classifiers. A novel ensemble model (EEXR) is proposed that combines three state-of-the-art CNN architectures, namely EfficientNet, Xception and ResNet152. Through extensive experiments on a Tamil medicinal plant dataset, EEXR shows the best performance with an accuracy of 96.71% and a lower error rate of 3.24% for the plant dataset.

References

[1] N. S. Jamir, H. K. Sharma, and A. K. Dolui, (1999), "Folklore medicinal plants of Nagaland, India", Fitoterapia, 70(4) 395-401.
[2] Y. Telwala, B. W. Brook, K. Manish, and M. K. Pandit, (2013), "Climate-induced elevational range shifts and increase in plant species richness in a Himalayan biodiversity epicentre", PloS one, 8(2), p. e57103.
[3] J. F. Maxwell, L. S. Pereira, and C. Padoch, Eds., (1994), "The social life of forest: The North East region of India", Lotus Press.
[4] M. Ekor, (2014), "The growing use of herbal medicines: issues relating to adverse reactions and challenges in monitoring safety", Frontiers in pharmacology, 4: 177.
[5] P. M. Unnikrishnan, (1998), "Awareness and practice of folk medicine in Travancore", in "Role of traditional medicine in primary medicare", P. Pushpangadan, Ed., IICH, Trivandrum, 153-158.
[6] Y. S. Malik et al., (2020), "Coronavirus disease pandemic (COVID-19): challenges and a global perspective", Pathogens, 9(6): 519.
[7] K. Abascal and E. Yarnell, (2004), "Herbal medicine for rheumatic diseases", Alternative and Complementary Therapies, 10(6):286-293.
[8] C. H. Saslis-Lagoudakis et al., (2012), "Phylogenies reveal predictive power of traditional medicine in bioprospecting", Proceedings of the National Academy of Sciences, 109(39):15835-15840.
[9] P. P. Aitwadkar, S. C. Deshpande, and A. V. Savant, (2019), "Identification of Indian Medicinal Plant by using Artificial Neural Network", International Research Journal of Engineering and Technology (IRJET), 5(4):1669-1671.
[10] B. R. Pushpa, C. Anand and P. Mithun Nambiar, (2016), "Ayurvedic Plant Species Recognition using Statistical Parameters on Leaf Images", International Journal of Applied Engineering Research, 11(7).
[11] D. Venkataraman and N. Mangayarkarasi, (2016), "Computer vision based feature extraction of leaves for identification of medicinal values of plants", in International Conference on Computational intelligence and Computing Research, 1-5.
[12] R. Janani and A. Gopal, (2013), "Identification of selected medicinal plant leaves using image features and ANN", in International Conference on Advanced Electronic Systems (ICAES), 1-5: 238-242.
[13] K. Meeta et al., (2012), "Survey on techniques for plant leaf classification", International Journal of Modern Engineering Research (IJMER), 1(2), pp. 538-544.
[14] P. Manojkumar, C. M. Surya, and V. P. Gopi, (2017), "Identification of Ayurvedic Medicinal Plants by Image Processing of Leaf Samples", in Third International Conference on Research in Computation Intelligence and Communication Networks,1-9.
[15] S. G. Wu, (2007), "A leaf recognition algorithm for plant classification using probabilistic neural network", in IEEE International Symposium on Signal Processing and Information Technology.
[16] S. Prasad, K. M. Kudiri, and R. C. Tripathi, (2013), "Relative Sub-Image Based Features for Leaf Recognition using Support Vector Machine", in Proceedings International Conference on Communication, Computing & Security,343-346.
[17] M. Keskar and D. Maktedar, (2019), "Enhancing Classifier Accuracy in Ayurvedic Medicinal Plants using WO-DNN", International Journal of Engineering and Advanced Technology (IJEAT), 9(1):19-25.
[18] A. Gokhale et al., (2020), "Computational Models for Identifying Indian Medicinal Plants", Journal of Critical Reviews, 7(1):1-9.
[19] S. Roopashree and J. Anitha, (2020), "Medicinal Leaf Dataset", Mendeley Data, V1, doi: 10.17632/nnytj2v3n5.1.