International Journal of Applied Engineering & Technology

DEEP LEARNING FOR MEDICINAL PLANT LEAF CLASSIFICATION: A HYBRID APPROACH

S Uma[1], Sandhya C[2] and V Padmapriya[3]
[1]Associate Professor, Department of Computer Applications, B.M.S. College of Engineering, Bangalore, Karnataka, India
[2]PG Student, Department of Computer Applications, B.M.S. College of Engineering, Bangalore, Karnataka, India
[3]Assistant Professor, Department of Computer Applications, B.M.S. College of Engineering, Bangalore, Karnataka, India
[1]uma.mca@bmsce.ac.in, [2]sandhyac.mca21@bmsce.ac.in and [3]padmapriyav.mca@bmsce.ac.in

ABSTRACT
Identifying medicinal leaf species is vital for preserving traditional knowledge and sustainable utilization of medicinal plants. Proper identification aids in distinguishing beneficial from toxic species, supporting pharmaceutical research, and conserving these vital plant resources and their habitats. This research addresses the critical task of accurately classifying medicinal plant leaves, with a dual focus on medical applications and ecological preservation. The primary objective of this study is to develop a robust deep learning model for precise leaf identification, harnessing India's rich biodiversity. The methodology involves the creation of a hybrid deep learning architecture that combines the strengths of Visual Geometry Group 16 (VGG16) and Densely Connected Convolutional Network 121 (DenseNet121) models. Indeed the achieved test accuracy includes the images from the testing set and the images sourced from the internet emphasizing the effectiveness and robustness of the proposed model. The novelty of this work lies in its comprehensive approach.

Index Terms - Classification, Deep Learning Model, DenseNet121 Models, Image Processing, Medicinal plant leaf, Transfer Learning, VGG16.

INTRODUCTION
India has a rich biodiversity, and plays a significant role in medicinal leaf usage, as it houses numerous medicinal plant species and contributes to the traditional medicine system. According to the report by the International Union for Conservation of Nature and the World Wildlife Fund, there are around 50,000 and 80,000 flowering plant species used for medicinal purposes worldwide [1]. These numbers emphasize the importance of preserving traditional knowledge. India hosts a diverse range of medicinal plant species, with some being widely used in households for their healing properties. The country's high usage of medicinal leaves can be attributed to its diverse ecosystem, which is conducive to the growth of various medicinal plants. The identification of leaf species is essential in the field of medicine and conservation of the species. Ancient civilizations relied on medicinal leaves for treating ailments, a practice that is still relevant today, such as the use of Tulsi leaves to cure coughs, etc. Medicinal leaves offer a holistic and sustainable approach to healthcare, containing bioactive compounds with multiple therapeutic benefits and minimal side effects when used appropriately. They also promote the preservation of indigenous knowledge and cultural heritage, strengthening communities.

Furthermore, medicinal leaves have environmental advantages over chemical-based medicines, requiring fewer resources for cultivation and contributing to biodiversity conservation. While chemical-based medicines have brought significant advancements in healthcare, they come with drawbacks like side effects, antibiotic resistance, and environmental pollution. Integrating traditional medicine with evidence-based practices harnesses the benefits of both medicinal leaves and modern medicine. Collaborative efforts between traditional healers, scientists, and healthcare practitioners can lead to standardized herbal remedies with proven efficacy and safety. The identification of medicinal plant leaves is essential for various reasons. Firstly, it enables the safe and effective utilization of these plants in indigenous healing practices and contemporary healthcare, ensuring that the right species are used for specific treatments. Secondly, accurate identification helps in distinguishing beneficial medicinal plants from potentially toxic ones. Additionally, it supports pharmaceutical research by providing a reliable source of medicinal plant materials.

Previous research has explored various methodologies for medicinal plant recognition using deep learning techniques such as convolutional neural networks (CNNs), transfer learning, and hyperparameter tuning. Datasets from diverse regions including Malaysia, Vietnam, and India have been utilized, comprising images of medicinal herbs and plant leaves. Different CNN architectures like VGG16, ResNet, InceptionV3, and MobileNet have been evaluated for their effectiveness in accurately identifying medicinal plants, showcasing the versatility and adaptability of deep learning approaches in this domain [2-9].

The proper identification of medicinal plant leaves is crucial for the conservation of these valuable plant resources and their natural habitats, contributing to biodiversity preservation and the sustainable use of these plants for future generations.

RELATED WORK

The precise identification of plant species through leaf classification is a crucial task with significant consequences for both healthcare and ecological preservation in the field of medicinal plant research. This literature review explores the advanced deep learning models created specifically for precise leaf identification. It highlights the importance of having robust, state-of-the-art models and in this survey, exploring recent research efforts, discussing the methodologies, accomplishments, and limitations.

In [10], a study employed a dataset comprising 2,400 leaf images from 40 species and applied a CNN model with three fully connected layers, achieving an accuracy of 95.06%. The model was compared to AlexNet and SVM, with SVM exhibiting an accuracy of 96.76%. However, this work includes limitations of small dataset size, the risk of overfitting, and the absence of testing on publicly available data. Authors in their work [11], have provided an extensive dataset comprising 64,000 leaf images from 64 species was utilized, achieving a CNN model accuracy of 95.79%. Notably, VGG16 and VGG19 models exhibited superior performance at 97.8% and 97.6% accuracy, respectively, following 25 epochs with a batch size of 16. However, a notable drawback is the large memory requirement when using ImageNet-based models due to their substantial number of parameters. Additionally, this study did not test its models on publicly available data.

In [12], a training process involving 37,693 leaf images from 10 species distributed across 3 batches, and various CNN architectures were evaluated. The results showed that the performance improved with increasing network depth: 2 layers achieved 62.80%, 3 layers reached 66.92%, 4 layers achieved 69.33%, and a more complex architecture with 3 layers, dropout, Gaussian noise, and batch normalization attained the highest accuracy at 71.3% over 300 epochs with a batch size of 64 and learning rate decay. However, a notable drawback is the extensive computational time required for training the CNN layers in three batches, approximately 7.5 hours. Additionally, this study didn’t test the model on publicly available datasets.

In [13], an experiment was carried out with 2,320 leaf images across 4 species, a CNN model comprising three sets of layers achieved results using K-fold cross-validation. In the second fold, the model exhibited a training accuracy of 100% and a testing accuracy of 87.73% after 25 epochs with a batch size of 32. However, a significant drawback lies in the dataset's limited diversity, comprising only four species, each with 75 samples. This restricted dataset size and diversity raise concerns of potential overfitting, as the model may excel on specific samples but struggle to generalize to new data. Moreover, the study did not assess the model's performance on publicly available data. In [14], a dataset of 13,500 leaf images representing 5 species, and a CNN model featuring a Global Average Pooling (GAP) layer was demonstrated. It achieved classification accuracy across various image resolutions: 99.66% at 64x64, 99.32% at 128x128, and 99.45% at 256x256 after 100 epochs of training. These results highlighted the effectiveness of GAP layers in feature extraction and classification. However, it's worth noting that the model's performance was not assessed on publicly available data.

In [15], a dataset containing 500 leaf images across 10 species, and four deep learning models, ResNet50, DenseNet201, VGG16, and InceptionV3 were employed for classification. The results showed varying levels of performance, with DenseNet201 achieving the highest accuracy at 97%, followed closely by VGG16 at 96%, and InceptionV3 at 95%, while ResNet50 performed slightly lower at 72%. These models were trained over 10 epochs with a batch size of 32. It's important to note that these pre-trained models, due to their depth and complexity, demand significant memory resources for both training and inference. Additionally, this study didn't test the model on publicly available datasets.

All the papers reviewed so far have common limitations, notably the absence of testing on publicly accessible datasets. However, the contribution of the authors in [16] is an exception. In this paper, the authors have used the Kaggle [17] dataset comprising 32,312 leaf images representing 30 different species of "Segmented Medical Leaf Images," several models were proposed for medicinal leaf identification. The models included CNN without pre-trained weights, VGG16, MobileNet, Xception, and InceptionResNetV2, all trained for 4 epochs. InceptionResNetV2 achieved the highest accuracy of 98.09% during training and a validation accuracy of 99.80%, outperforming the other models. However, this study lacked comprehensive details regarding hyperparameters and augmentation techniques. Specifically, there were insufficient descriptions of the data augmentation techniques employed, the number of internet images used, and the absence of graphical representations of accuracy on the testing set. Nevertheless, it's important to highlight that this study was conducted using a dataset that is publicly accessible.

The methodology combines VGG16 and DenseNet121, creating a hybrid deep-learning model. This fusion maximizes feature extraction capabilities, capturing both shallow and deep image features for a comprehensive representation.

METHODOLOGY

The architectural design of the proposed deep learning model [18] developed for the classification of medicinal plant leaves includes several phases. The proposed model combines two powerful pre-trained models, VGG16 [19], [20] and DenseNet121 [21], [22] to maximize its feature extraction capabilities. The dataset is split into three distinct sets: a training set containing 70% of the images, a validation set with 10% of the images, and a testing set comprising 20% of the images. In the testing set, five internet images are added for each leaf species to enrich the test set. For each leaf species in the testing set, five internet photos are added to enrich the test set.

The following steps are part of the deep learning model training process:

Model Training Process:

The model training process involves the following phases:

1. **Dataset Loading and Encoding:** Images are resized to (128*128), and labels are loaded and prepared for processing. Then, label encoding and one-hot encoding are used to convert the categorical class labels into a numerical format.
2. **Pre-trained Models Setup:** The VGG16 and DenseNet121 models are loaded with weights trained on a large image dataset. Then, all layers of both models are frozen to retain the features learned from initial training, and the outputs are flattened to create a feature vector individually. Then, the features are concatenated to form a hybrid model of VGG16 and DenseNet121 [23].
3. **Fully Connected Layers:** The model includes a series of dense layers for classification. Five fully connected layers are added after the feature fusion. The number of neurons in these layers gradually decreases, from 1024 to 512, 256, 128, and finally 64. Each dense layer is followed by a Rectified Linear Unit (ReLU) activation function. ReLU introduces non-linearity to the model, allowing it to learn complex patterns and relationships within the data. Batch normalization is applied after the dense layer, and a dropout rate of 0.2 is used after the fifth dense layer to enhance model robustness [24].
4. **Learning Rate Scheduler:** A learning rate scheduler function is defined based on the epoch. The learning rate starts high and decreases over time, which can help the model converge better. The model is trained for 11 epochs, initially using a higher learning rate of 0.0001 for the first 5 epochs to allow the model to make larger

updates to the weights. Then, the learning rate is reduced to 0.00001 for the next 5 epochs to fine-tune the model with smaller updates. Finally, a very small learning rate of 0.000001 is used for the last epoch [25].

5. **Fine Tuning and Training the model:** Layers are unfrozen after the 10th layer in VGG16 and layers after the 50th layer in DenseNet121 to allow them to be updated during training for fine-tuning. The model is then trained with a batch size of 128 and 11 epochs. The output layer of the model employs the softmax activation function. The model is compiled using categorical cross-entropy as the loss function, which is commonly used for multi-class classification problems. The Adam optimizer is utilized for efficient and adaptive gradient-based optimization during training.

Dataset:

The "Segmented Medical Leaf Images" dataset which was obtained from Kaggle and is publicly available, consists of 1,835 images representing 30 distinct medicinal plant species. TABLE I list the names of 30 different medicinal plant species included in the dataset, each identified by its scientific botanical name and common name.

Table I. Medicinal Plant Species Names

Segmented Medicinal Leaf Images	
1.Alpinia Galanga (Rasna)	16.MuntingiaCalabura (Jamaica Cherry-Gasagase)
2.Amaranthus Viridis (Arive-Dantu)	17.MurrayaKoenigii (Curry)
3.Artocarpus Heterophyllus (Jackfruit)	18.Nerium Oleander (Oleander)
4.Azadirachta Indica (Neem)	19.Nyctanthes Arbor-tristis (Parijata)
5.Basella Alba (Basale)	20.OcimumTenuiflorum (Tulsi)
6.Brassica Juncea (Indian Mustard)	21.Piper Betle (Betel)
7.Carissa Carandas (Karanda)	22.PlectranthusAmboinicus (Mexican Mint)
8.Citrus Limon (Lemon)	23.Pongamia Pinnata (Indian Beech)
9.Ficus Auriculata (Roxburgh fig)	24.Psidium Guajava (Guava)
10.Ficus Religiosa (Peepal Tree)	25.Punica Granatum (Pomegranate)
11.Hibiscus Rosa-sinensis	26.Santalum Album (Sandalwood)
12.Jasminum (Jasmine)	27.SyzygiumCumini (Jamun)
13.Mangifera Indica (Mango)	28.SyzygiumJambos (Rose Apple)
14.Mentha (Mint)	29.TabernaemontanaDivaricata (Crape Jasmine)
15.Moringa	30.Trigonella Foenum-

Segmented Medicinal Leaf Images	
Oleifera (Drumstick)	graecum (Fenugreek)

Image Processing Techniques:

To enhance the dataset, various data augmentation methods [26] were applied, such as rotation, flipping, cropping, and scaling, as well as adjustments to brightness and contrast. This augmentation process resulted in a total of 31,344 images. Slight random variations are applied to the original image to introduce subtle changes. These variations help the model adapt to different orientations since images may be captured at various angles or orientations. The dataset was enhanced using the following techniques, which are listed below:

1. **Cropping:** Crop operation is applied on each original image, where a specific rectangular region is chosen, spanning from (50, 50) to (150, 150) pixels, effectively focusing on a central portion of the image, helps the model to concentrate on the most important features for classification.
2. **Padding:** A border of 10 pixels is added to all sides of the image, which helps maintain the spatial dimensions of the feature map.
3. **Flipping:** Horizontal flipping is applied to create mirror images.
4. **Translations:** Two translations (10, 10) and (-10, 10) pixel quantities are applied to images in both the horizontal and vertical directions to generate slight positional adjustments. By using these values (10, 10) and (-10, -10), it is possible to see a visible shift in the original image.
5. **Rotations:** The images are rotated by angles of 0, 90, 180, and 270 degrees, providing a wide range of orientation variations for the augmented images. The original image is rotated using these values in each of the four directions.
6. **Scaling:** Images are resized using scale factors of 0.8, 1.0, and 1.2 to introduce variability in the dataset without making the images too small or too large. The original image is somewhat altered by being zoomed out and in by 0.8 and 1.2, respectively.
7. **Shear:** Shearing is applied to the image using shear factors of 0.1 and 0.2.To improve a model's ability to handle photos with various degrees of tilt or skew, a shearing factor of minimal values 0.1 and 0.2 is applied to detect changes in the original images.
8. **Zooming:** Images are zoomed in using factors of 1.2 and 1.5. Moderate values 1.2 and 1.5 are selected to have variation in the dataset.
9. **Color Shift:** Random shift values between 20 and 50 are added to the pixel values and are applied to each color channel (BGR) to change the color appearance of the images.
10. **Noise Addition:** Gaussian noise is added with a mean of 0 and a scale of 10 is added to the images using a normal distribution.
11. **Elastic Deformation:** Elastic deformation is applied to images by distorting them based on random shifts calculated using a random state.

RESULTS AND DISCUSSIONS

The combination of two transfer learning architectures, VGG16 and DenseNet121, is trained with a defined learning rate scheduler. Figure 1 provides the result of the model's performance across multiple training epochs.

```
Epoch 8/11
170/170 [==============================] - 65s 381ms/step - loss: 0.0362 - accuracy: 0.9907 - val_loss: 0.0307 - val_accuracy:
0.9898 - lr: 1.0000e-05
Epoch 9/11
170/170 [==============================] - 65s 379ms/step - loss: 0.0341 - accuracy: 0.9928 - val_loss: 0.0279 - val_accuracy:
0.9933 - lr: 1.0000e-05
Epoch 10/11
170/170 [==============================] - 66s 385ms/step - loss: 0.0313 - accuracy: 0.9930 - val_loss: 0.0300 - val_accuracy:
0.9901 - lr: 1.0000e-05
Epoch 11/11
170/170 [==============================] - 65s 382ms/step - loss: 0.0282 - accuracy: 0.9942 - val_loss: 0.0274 - val_accuracy:
0.9911 - lr: 1.0000e-06
200/200 [==============================] - 10s 52ms/step - loss: 0.1890 - accuracy: 0.9704
Test Accuracy: 0.9704271554946899
```

Figure 1: The Result of the Model's Performance, Showing the Loss, Accuracy, and Learning Rate Values for Each Epoch

A batch size of 128 is utilized for each iteration, and the training process is conducted over 11 epochs. The model's performance is evaluated using the validation dataset during training. Finally, the model's accuracy is evaluated on the test dataset, including internet images that were not part of the training data. Further, the results illustrate the model's training accuracy at 99.42% and validation accuracy at 99.11%, with a final testing dataset accuracy of 97.04%.

The training accuracy of 99.42% and validation accuracy of 99.11% indicate that the model has learned to classify the training data with high accuracy and can also generalize well to new, unseen data. As illustrated in Figure 2, the proximity of the training and validation accuracy curves is a positive sign, indicating that the model is not experiencing significant overfitting.

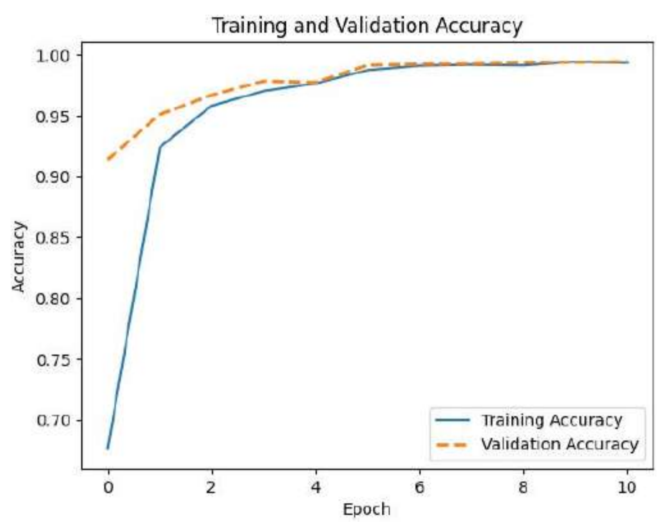

Figure 2. Training and Validation Accuracy of VGG16 and DenseNet121 model

Figure 3 emphasizes that both training and validation losses are consistently decreasing, which indicates that the model is learning from the data and improving its predictions. The validation loss remains relatively close to the training loss, suggesting that the model is generalizing well to new data.

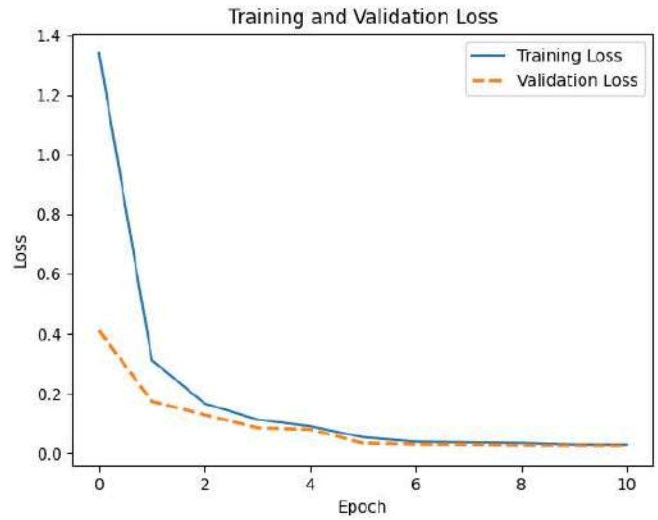

Figure 3. Training and Validation Loss of VGG16 and DenseNet121 model

Figure 4 illustrates the model's robust performance with a validation accuracy of 99.11% and a test accuracy of 97.04%, demonstrating its ability on both seen and unseen data. This indicates that it has successfully learned meaningful features from the training data and can effectively generalize to new instances.

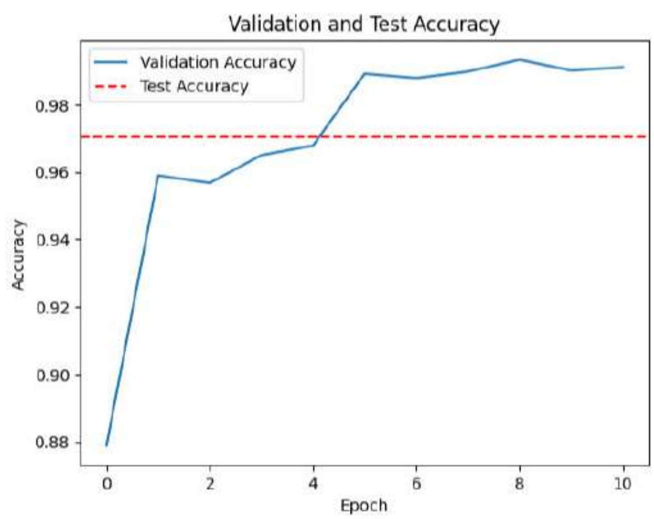

Figure 4: Validation and Test Accuracy of VGG16 and DenseNet121 model

The confusion matrix helps to understand the performance of a classification model. Figure 5 depicts the confusion matrix for all 30 classes.

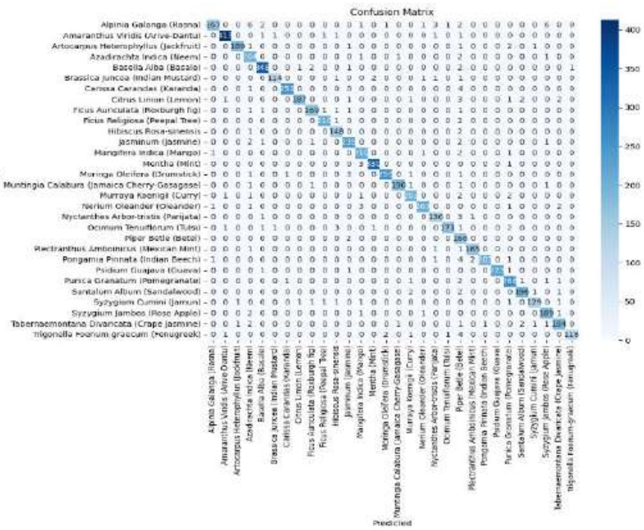

Figure 5: Confusion Matrix

However, Table II depicts the confusion matrix for medicinal plants with 30 classes. It consists of predicted class, actual class, and the count of instances for each scenario. In this matrix, there are 6190 instances where the predicted class matches the actual class, representing true positives (TP) for each class. The sum of these values indicates the total number of correct classifications made by the model.

Additionally, there are 201 instances where the predicted class does not match the actual class, indicating the total number of incorrect classifications. Total Incorrect (201) indicates that there were 201 instances in the test dataset that the model misclassified. This could be due to the complexity of the images, similarities between certain classes, or limitations in the model's ability to generalize.

Table II. Confusion Matrix of Medicinal Plant with 30 Classes

	Predicted Class	**Actual Class**	**Count**
0	Total Correct	Total Correct	6190
1	Total Incorrect	Total Incorrect	201

In Table III, the confusion matrix provides an overview of the model's performance for two classes, 'AlpiniaGalanga (Rasna)' (class 0) and 'AmaranthusViridis (Arive-Dantu)' (class 1). For the class - 0 "AlpiniaGalanga (Rasna)", the model correctly predicts it as "AlpiniaGalanga (Rasna)" in 167 cases. There were no cases where the model incorrectly predicted other classes as "AlpiniaGalanga (Rasna)".For class - 1 "AmaranthusViridis (Arive-Dantu)", the model correctly predicts it as "AmaranthusViridis (Arive-Dantu)" in 413 cases. There were no cases where the model incorrectly predicted other classes as "AmaranthusViridis (Arive-Dantu)".

Table III. Confusion Matrix Overview of Medicinal Plant

Confusion Matrix Overview		
	Predicted: 1	Predicted: 0
Actual: 1	413	0
Actual: 0	0	167

The study's results demonstrate the effectiveness of the deep-learning model, which combines VGG16 and DenseNet121 architectures. With a training accuracy of 99.42% and a validation accuracy of 99.11%, the model exhibits strong performance on both seen and unseen data, indicating its ability to generalize well. The test accuracy of 97.04% further emphasizes its capability to classify medicinal plant leaves accurately, even when exposed to internet-sourced images not part of the training data. The analysis of the confusion matrix reveals that, although there were some misclassifications, the model's overall efficiency and its ability to accurately classify various medicinal plant species remain robust.

CONCLUSION

India's rich biodiversity, housing numerous medicinal plant species, emphasizes the need for precise leaf identification. A novel deep learning approach is presented for the accurate classification of medicinal plant leaves, a task of significant importance for both medicinal applications and ecological preservation. This study introduces a robust deep learning model using publicly available data, addressing existing challenges. The hybrid deep learning model combines VGG16 and DenseNet121, enhancing leaf classification accuracy. Augmenting the "Segmented Medical Leaf Images" dataset and incorporating internet images demonstrates the model's robustness, achieving 99.42% training and 99.11% validation accuracy. This model addresses limitations identified in previous research and provides insights into data augmentation, graphical accuracy representation, and detailed hyperparameter settings. Notably, it achieves 97.04% test accuracy on unseen internet images. In future work, the model's capabilities can be extended to identify leaves present in groups or clusters, and identification of leaves with diverse background conditions, and a mobile application can be developed for real-time leaf identification, further bridging the gap between traditional knowledge and modern science in the realm of medicinal plants.

REFERENCES

[1] "Medicinal Plants: An Overview," International Union for Conservation of Nature (IUCN): https://www.iucn.org/resources/
issues-briefs/medicinal-plants-overview.

[2] Mohd Roslan, Noor Aini, Norizan Mat Diah, Zaidah Ibrahim, Yuda Munarko, and Agus Eko Minarno. "Automatic plant recognition using convolutional neural network on Malaysian medicinal herbs: the value of data augmentation." *International Journal of Advances in Intelligent Informatics* 9, no. 1 (2023).

[3] Amulya, K., K. Deepika, and P. Kamakshi. "An optimized hyper parameter-based CNN approach for predicting medicinal or non-medicinal leaves." Advances in Engineering Software 172 (2022): 103181. Tiwari V, Joshi RC, Dutta MK. Deep neural network for multi-class classification of medicinal plant leaves. Expert systems. 2022 Sep; 39(8):e13041.

[4] Abdollahi, Jafar. "Identification of medicinal plants in ardabil using deep learning: identification of medicinal plants using deep learning." In 2022 27th International Computer Conference, Computer Society of Iran (CSICC), pp. 1-6. IEEE, 2022.

[5] Quoc, Trung Nguyen, and Vinh Truong Hoang. "Medicinal Plant identification in the wild by using CNN." In 2020 International Conference on Information and Communication Technology Convergence (ICTC), pp. 25-29. IEEE, 2020.

[6] TS, Sai Kumar, and A. Prabalakshmi. "Identification of indian medicinal plants from leaves using transfer learning approach." In 2021 5th International Conference on Trends in Electronics and Informatics (ICOEI), pp. 980-987. IEEE, 2021.

[7] Geerthana, R., P. Nandhini, and R. Suriyakala. "Medicinal Plant Identification Using Deep Learning." International Research Journal on Advanced Science Hub 3 (2021): 48-53.

[8] Pushpa, B. R., and N. S. Rani. "Ayur-PlantNet: An unbiased light weight deep convolutional neural network for Indian Ayurvedic plant species classification." Journal of Applied Research on Medicinal and Aromatic Plants 34 (2023): 100459.

[9] Dileep MR, Pournami PN. AyurLeaf: a deep learning approach for classification of medicinal plants. InTENCON 2019-2019 IEEE Region 10 Conference (TENCON) 2019 Oct 17 (pp. 321-325). IEEE.

[10] Paulson, Anu, and S. Ravishankar. "AI based indigenous medicinal plant identification." In 2020 Advanced Computing and Communication Technologies for High Performance Applications (ACCTHPA), pp. 57-63. IEEE, 2020.

[11] Akter, Raisa, and Md Imran Hosen. "CNN-based leaf image classification for Bangladeshi medicinal plant recognition." In 2020 Emerging Technology in Computing, Communication and Electronics (ETCCE), pp. 1-6. IEEE, 2020.

[12] Latumakulita, Luther, Franklin Mandagi, Frangky Paat, Dedie Tooy, Sandra Pakasi, Sofia Wantasen, Diane Pioh et al. "Web-based system for medicinal plants identification using convolutional neural network." Bulletin of Social Informatics Theory and Application 6, no. 2 (2022): 158-167.

[13] Azadnia, Rahim, Mohammed Maitham Al-Amidi, Hamed Mohammadi, Mehmet Akif Cifci, Avat Daryab, and Eugenio Cavallo. "An AI Based Approach for Medicinal Plant Identification Using Deep CNN Based on Global Average Pooling." Agronomy 12, no. 11 (2022): 2723.

[14] Borkatulla, Bijly, Jannatul Ferdous, Abdul Hasib Uddin, and Prince Mahmud. "Bangladeshi medicinal plant dataset." Data in Brief 48 (2023): 109211.

[15] Rani, Leena, G. Devika, Asha Gowda Karegowda, S. Vidya, and Shreetha Bhat. "Identification of medicinal leaves using state of art deep learning techniques." In 2022 IEEE International Conference on Distributed Computing and Electrical Circuits and Electronics (ICDCECE), pp. 1-5. IEEE, 2022.

[16] [Available Online]: https://www.kaggle.com/datasets/riteshranjansaroj/segmented-medicinal-leaf-images

[17] Krizhevsky, Alex, Ilya Sutskever, and Geoffrey E. Hinton. "Imagenet classification with deep convolutional neural networks." Advances in neural information processing systems 25 (2012).

[18] Simonyan, Karen, and Andrew Zisserman. "Very deep convolutional networks for large-scale image recognition." arXiv preprint arXiv: 1409.1556 (2014).

[19] He, Kaiming, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. "Deep residual learning for image recognition." In Proceedings of the IEEE conference on Computer Vision and pattern recognition, pp. 770-778. 2016.

[20] Huang, Gao, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q. Weinberger. "Densely connected convolutional networks." In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 4700-4708. 2017.

[21] Shaha, Manali, and Meenakshi Pawar. "Transfer learning for image classification." In 2018 second international conference on electronics, communication and aerospace technology (ICECA), pp. 656-660. IEEE, 2018.

[22] Qiu, Xipeng, Tianxiang Sun, Yige Xu, Yunfan Shao, Ning Dai, and Xuanjing Huang. "Pre-trained models for natural language processing: A survey." Science China Technological Sciences 63, no. 10 (2020): 1872-1897.

[23] Ma, Wei, and Jun Lu. "An equivalence of fully connected layer and convolutional layer." arXiv preprint arXiv: 1712.01252 (2017).

[24] Yedida, Rahul, and Snehanshu Saha. "A novel adaptive learning rate scheduler for deep neural networks." arXiv preprint arXiv: 1902.07399 (2019).

[25] Xu, Mingle, Sook Yoon, Alvaro Fuentes, and Dong Sun Park. "A comprehensive survey of image augmentation techniques for deep learning." Pattern Recognition (2023): 109347.

[26] Heydarian, Mohammadreza, Thomas E. Doyle, and Reza Samavi. "MLCM: Multi-label confusion matrix." IEEE Access 10 (2022): 19083-19095.