Plumbago Zeylanica Leaf Classification using CNNs and Transfer Learning

Md.Rabbi
Dept. of CSE
East West University
Aftabnagar,Dhaka,Bangladesh
Email: rabbim913@gmail.com

Saheba Akter
Dept. of CSE
East West University
Aftabnagar,Dhaka,Bangladesh
Email: saheba188416@gmail.com

***Abstract*—This study focuses on the classification of a medicinal plant Plumbago Zeylanica (Chitrak). A machine learning-based system has been implemented to classify *Plumbago Zeylanica* (Chitrak) into "Dried", "Healthy", and "Unhealthy" categories using 9,416 photos taken during the course of development. Classifiers were trained using convolutional neural networks, including a Custom CNN, VGG19, EfficientNet_B3, EfficientNet_B6, DenseNet201, and ResNet152. Data augmentation was used to enhance the performance of the model.The performance and presentation of model predictions were made understandable using XAI tools such as Grad-CAM, Grad-CAM++, Eigen-CAM, and LIME. This provided important information about how the decisions were arrived at. Among these models , Efficient_B3 provides highest accuracy of 99.89%, while *VGG19* provides the lowest accuracy of 89.92%, which is also good enough. The model's decision-making processes were validated by the Grad-CAM visualizations, which verified that they concentrated on affected regions. This research shows how effective it is to classify leaf classification using explainable AI approaches in conjunction with custom and transfer learning models.**

***Index Terms*—Plumbago Zeylanica, Custom CNN, VGG19, EfficientNet_B3, EfficientNet_B6, DenseNet201, ResNet152, Grad-CAM, Grad-CAM++, Eigen-CAM, LIME**

I. INTRODUCTION

Plants have been used for medical purposes by different civilizations since ancient times until today [10]. The World Health Organization(WHO) states that around 21,000 plants have been utilized for medicinal purposes, and according to the Food and Agricultural Organization(FAO), over 50,000 plant species are employed in traditional medicine worldwide [9] . Accurate identification of these plants is crucial for ensuring quality full and efficient medicine. Traditional methods for identification depend on manual inspection, which is time-consuming and also not accurate. Advanced technologies are making their way out to classify plant leaves fast accurately. Deep learning approaches, especially Convolutional Neural Networks(CNNs), have obtained tremendous success in image recognition [11].

The current research is on *Plumbago Zeylanica*(Chitrak) leaf classification. *Plumbago Zeylanica*, often known as chitrak, is valued for its medicinal properties.While there are many classification methods, there are few that work well for medicinal plants like chitrak. The interpretability problems with existing approaches make it challenging to understand the decision-making process in medical and therapeutic situations.

A custom Convolutional Neural Network(CNN) architecture was tailored for the classification of *Plumbago Zeylanica*(Chitrak) leaf images, addressing the unique morphological features of the species. Five pretrained models VGG19, EfficientNet_B3, EfficientNet_B6, DenseNet201 and Resnet152 were applied to the leaf dataset to evaluate their efficiency in classification. Among these, EfficientNet_B3 provides the highest accuracy of 99.89%. Significant areas of images have been highlighted by applying Gradient-weighted class activation mapping,it also ensures transparency and more accuracy. The comparative analysis of accuracy, precision ,recall and F1-score ensures the best fit model to classify these images.

This work provides a comprehensive approach for categorizing Plumbago Zeylanica (Chitrak) leaves using deep learning. This project also emphasizes reproducibility through the use of a well-documented Kaggle notebook. By combining explainability and performance, our study provides a scalable method for classifying plants that might be used for studies of medicinal and botanical plants.

II. RELATED WORK

Recent research indicates that deep learning—more specifically, Convolutional Neural Networks, or CNNs—can be used to correctly classify plant leaves. Amit Dhoka et al. presented a dataset on the medicinal leaf Plumbago Zeylanica (Chitrak) [1].

These datasets perform exceptionally well in the classification of leaves. For example, Althuniyan [2] introduced the CNN-based system "deepleaf," which uses efficient preprocessing and data augmentation approaches to automate leaf categorization with high accuracy. Additionally, transfer learning has significantly improved categorization performance. Siddharth [3].achieved an exceptional 99.70% accuracy rate in identifying leaf pictures by combining a transfer learning model with a pre-trained VGG-19 model. Similarly, Attallah [4] used compact CNNs with transfer learning and feature selection to classify tomato leaf diseases with high accuracy rates. using databases like Swedish Leaf and Flavia. Lee [5] used Xception, ResNet-59, DenseNet-201, and Vision Transformer to develop a 100% correct model. Kirubasri Gopalan [6] recently showed that the ResNet152 model, which has been enhanced with Grad-CAM for explainability, can detect maize leaf disease and produce intelligible answers. The accuracy of the algorithms used by Rodrigo [7] to identify photos of rice leaf disease using MobileViTV2_050 was 99.6%. Epie Custodio [8] has done a medicinal plant classification, getting 98.22% by averaging ensembles of deep learning convolutional neural networks.

Since many models do not offer interpretive predictions, it is difficult to understand the decision-making process. The study's low potential and variety have an influence on generalizability. Furthermore, since many therapeutic plants, like Plumbago Zeylanica, have not been studied, there are no exact categorization models.

We propose a method for classifying Plumbago Zeylanica leaves using five pre-trained models and a proprietary CNN in light of these research gaps. We enhanced interpretability and provided information on model selections using Grad-CAM images. Because our data was sourced from Mendeley Data, it is certain to be diverse and relevant to the target species.

III. METHODOLOGY

A. Structure of Custom CNN

The objective of the study was to identify the efficiency of the CNN model in classifying photos of plumbago zeylanica obtained from natural environments. In support of the objectives of this study, the section explains data, data preprocessing, splitting the dataset, designing a model, the evaluation of models, and then visualizing data by XAI and LIME. We can see the flowchart of methodology in Fig-1.

The architecture of the modified CNN was designed to gradually learn spatial hierarchies from input images of leaves. The network can efficiently learn low- to high-level features by beginning with 32 filters and gradually increasing to 128 across convolutional layers while maintaining computational efficiency. The 3*3 kernel size with ReLU activation ensures nonlinearity and localized pattern recognition. MaxPooling layers control overfitting and reduce spatial dimensions by using 2*2 kernels. Dropout after Dense Layers, which randomly disables a subset of neurons, is used for regularization to prevent overfitting and promote generalization. Additionally, batch normalization stabilizes learning, improves robustness, and accelerates convergence by normalizing inputs to each layer.

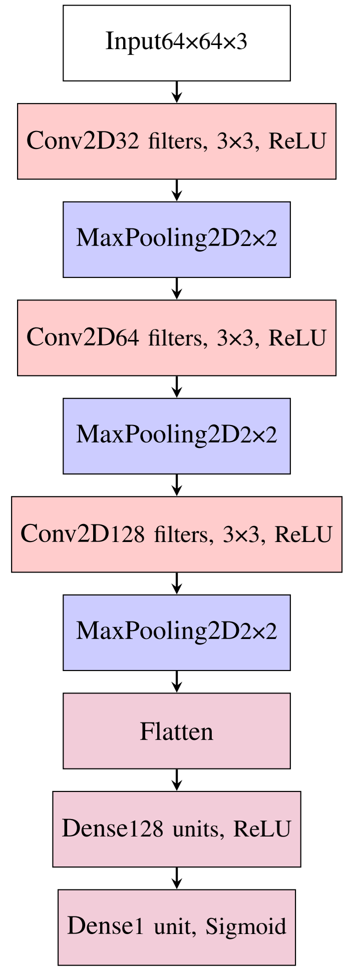

Fig. 1. Architecture diagram of the custom Convolutional Neural Network (CNN) showing each layer, kernel size, activation function, and pooling.

B. Transfer Learning

In this study, five pre-trained models- VGG19, EfficientNet_B3, EfficientNet_B6, DenseNet201 ResNet152 were employed to classify images of medicinal plant leaves. All models were utilized with ImageNet weights. For VGG19, the convolutional base was frozen and a new classification head with Flatten, Dense(256, ReLU), Dropout(0.5), and a softmax output layer was appended. EfficientNet_B3 EfficientNe_B6 employed GlobalAveragePooling2D filled by Dense layers(256 512 neurons respectively), ReLU activations, Dropout(0.4-0.5), and softmax classification layers, with top layers selectively unfrozen during fine-tuning. ResNet152 retained lower residual blocks frozen, while higher layers were unfrozen for transfer learning, and a classification head with GlobalAveragePooling2D, Dense(256, ReLU), Dropout(0.5), and softmax was added. DenseNet201, known for its densely connected layers, followed a similar strategy, incorporating a custom head with GlobalAveragePooling2D, Dense(256, ReLU), Dropout(0.5), and softmax . Each model was fine-tuned using the Adam optimizer with learning rates between 1e-5 and 1e-4, batch sizes of 32, and callbacks like EarlyStopping to avoid overfitting. This multi-model approach enabled a comprehensive evaluation of performance and interpretability in the medicinal plant classification task.

IV. Dataset & Preprocessing

A. Dataset Description

For deep learning models to be successfully trained and evaluated, especially for classification tasks, a large dataset is essential. We used a dataset of **Plumbago Zeylanica** (Chitrak) from our course instructor, which we obtained via Mendeley [1], for this study. This dataset is licensed under CC BY 4.0. It has a total number of 9416 images and three classes. Three classes are Dried, Healthy, Unhealthy containing 2029, 5088 and 2099 images respectively. We splitted the dataset into train, test and validation.

TABLE I
Dataset Partitioning Details

Partition	Percentage	Image Count
Train	70%	6591
Validation	20%	1883
Test	10%	942

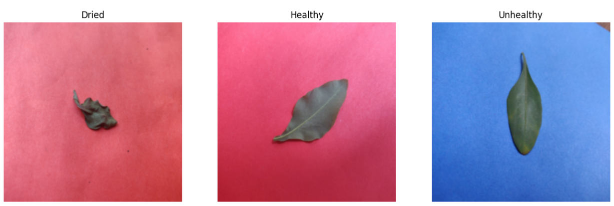

Fig. 2. Images of each class from dataset.

B. Data Processing and Augmentation

The dataset was preprocessed and augmented first to prepare for the training phase of our proposed deep learning systems. All images are resized to the same size 128 × 128 pixels, as a way to reduce computational processing. Data augmentations like standard flipping, random rotation of 20%, brightness adjustment of 20%, saturation adjustment of 20%, and contrast improvement of 20% were performed to enhance variance in the image dataset. Our data set has three classes-dried,healthy and unhealthy.Each class contains respectively 2029, 5088, 2099 images. As these augmentations simulate real variations in camera position, light, and environmental exposure during the image capture, much more diversity was brought forth into this dataset. This expanded not only the size but also helped to reduce the dimensionality complexity, thus benefiting the model in terms of performance and generalization. Fig.2 is the sample images of each class from the dataset.

V. Experimental Setup

A. Hardware and Software

All the training and testing methods were carried out in the kaggle notebook which provided us with a cloud based environment. All experiments were run in a kaggle notebook using python using T4 X 2 GPU and upto 16 GB RAM. Here we use many software and libraries like os, numpy, matplotlib, pytorch, tensorflow etc.

B. Training Detalis

The supervised learning approach used in the plumbago zeylanica stage classification training process ensured accurate and consistent results. Fig.2 shows how this process was carefully carried out in an organized manner. Below is a summary of the steps that make up the training methodology:
a) Data Splitting: Three subsets—training, validation, and testing sets—were methodically created from the dataset. The training set contained 70% of the entire dataset, with the remaining 20% being for validation and 10% for testing sets. Strong training and efficient assessment were guaranteed by this method. Our dataset has a total of 9416 annotated photos, for instance, 6591 images of all dried, healthy and unhealthy stages were used for training, and 1883 images were saved for validation and the remaining 942 images were used for testing.
b) Hyperparameters: Hyperparameters affect the architecture, functionality and overall performance of machine learning models. To get the best results you need to tune these.
c) Learning Rate: The learning rate is a hyperparameter which controls how much the model's weights are updated during training. In our study, we have observed that a higher learning rate gives an overfitted graph while a very lower learning rate gives an underfitted graph. With Adam optimizer in all models for efficient weight updates and consistent performance, we used a learning rate of 0.001 for Custom CNN and EfficientNet_B3 and for the Vgg19, we used a learning rate of 0.00001.

d) Regularization: For fine tuning, we implemented dropout and batch normalization regularization in Efficient Net_B6. By using the Adam optimizer, we passed the weight decay parameter with the value set to 1e-6. This approach helped our model to generalize better by preventing from learning large weights in training, and prevented the model from overfitting as well.
e) Batch Size: The batch size which is the number of samples processed before the model updates its parameters was set to 32 for Custom CNN, VGG19,EfficientNet_B3, EfficientNet_B6, DenseNet201, and ResNet152 in training, validation and testing. This batch size provides a good balance between computational efficiency and model generalization.
f) Number of Epochs: The maximum number of training epochs for any CNN model was set to 50. Early stopping was used to prevent overfitting and to save computing cost. The training would stop for 5 consecutive epochs if the validation loss didn't improve, so the models converged well and not overtrained.
g) Efficient Computation: We have integrated Autocast into our model to perform efficient computation by utilizing the GPU for accelerating the training process with reduced memory usage while maintaining model accuracy. Autocast is a pytorch feature that allows Automatic Mixed Precision (AMP) training. It converts the 32 bit floating point operations to 16 bit floating point to reduce the memory and faster the

computational speed.

C. Evaluation Matrics

The criteria outlined in our study document must be followed to evaluate the efficacy of our models. The performance of our models is evaluated using a variety of evaluation indicators.

*1) **Accuracy**:*

Accuracy is one of the most important criteria used to assess the performance of our models. This metric shows the proportion of accurate predictions the models produced. The following formula is used to determine accuracy:

$$\text{Accuracy} = \frac{TP + TN}{\text{Total}} \tag{1}$$

It is impossible to completely describe the models' overall performance in the context of our paper without accuracy metrics. With respect to our models, the accuracy gives information about how well the model can classify occurrences.

*2) **Precision**:*

Precision measures the degree to which our models produce favorable predictions. It is important because it helps us to prevent false positives in our models. It also shows the relevance of the favorable cases anticipated. Calculating precision is as follows:

$$\text{Precision} = \frac{TP}{TP + FP} \tag{2}$$

*3) **Recall**:*

Recall measures how often the machine learning model accurately detects real-world occurrences. It is crucial to understand genuine positive values since we are developing a model to classify leaf diseases. One way to calculate recall is to:

$$\text{Recall} = \frac{TP}{TP + FN} \tag{3}$$

*4) **F1-Score**:*

In particular, the F1-Score—the harmonic mean of Precision and Recall—is relevant to our research. By achieving a balanced combination of precision and recall in every category, we hope to create a reliable classification system. The F1-Score is computed by:

$$\text{F1-Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{4}$$

VI. Result

To enhance the performance of the models, we incorporated a total of 6591 images for the training and a total of 1883 images for validation stages. In addition, the model's performance on the 942 images was also thoroughly assessed to understand its generalization capabilities. In Fig.3, we observe that, from the training accuracy results of the models, the highest scores were achieved by EfficientNet_B3 - 99.89%, and the other models accuracy scores are Custom CNN - 98.41%, VGG19 - 89.92%, EfficientNet_B6- 99.58% , DenseNet201 - 99.83%, and ResNet152 - 96.71%. The outlined results confirm the efficiency of the models and their relevance to the task set.

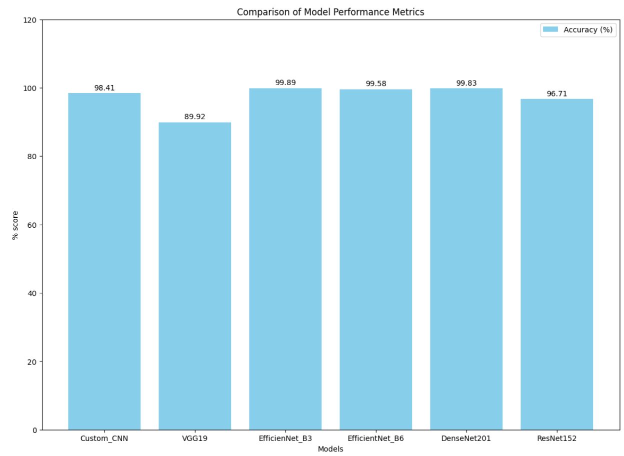

Fig. 3. Comaparison between Models

The data in Figures 4, 6, 8, 10, 12, and 14 are related to loss curves that corroborate the validation of these models' effectiveness. These images clearly show the convergence behaviour and stability of all models during the training, as well as their training and validation losses. The confusion matrices presented in Figures 5, 7, 9, 11, 13, and 15 depict the classification performances of our models, namely Custom CNN, VGG19, Efficient_B3, EfficientNet_B6, DenseNet201, and ResNet152 respectively,for distinguishing between dried, healthy and unhealthy leaves of **Plumbago Zeylanica** (Chitrak).

TABLE II
PERFORMANCE METRICS OF TRAINED MODELS

Model	Accuracy	Precision	Recall	F1-Score
Custom CNN	98.41%	0.9841	0.9841	0.9840
VGG19	89.92%	0.9028	0.8992	0.9004
EfficientNet_B3	99.89%	0.9989	0.9989	0.9989
EfficientNet_B6	99.58%	0.9958	0.9958	0.9958
DenseNet201	98.83%	0.9886	0.9883	0.9883
ResNet152	96.71%	0.9685	0.9671	0.9674

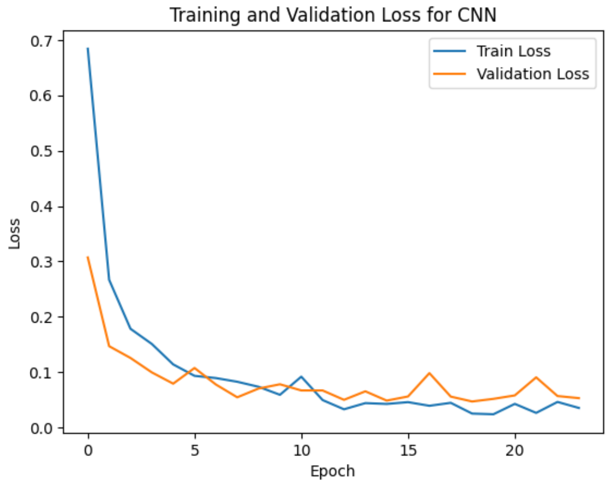

Fig. 4. Loss Curve for Custom CNN demonstrates that both training and validation losses quickly drop and level out at low values, signifying successful learning. Good generalization without overfitting is suggested by the two curves' near alignment.

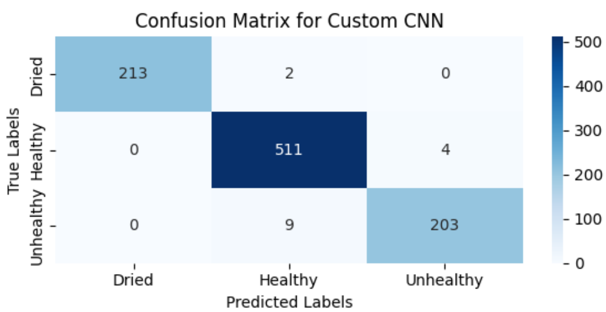

Fig. 5. Confusion Matrix for Custom CNN

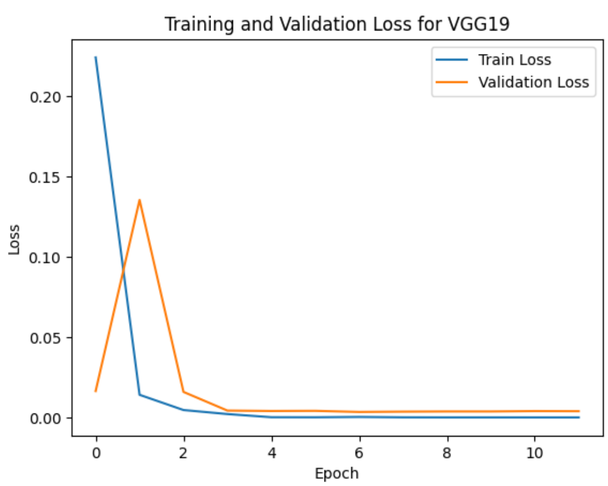

Fig. 6. Loss Curve for VGG19 shows a rapid drop in both training and validation losses within the first few epochs, followed by the early convergence to near-zero values. This indicates efficient learning and excellent model generalization.

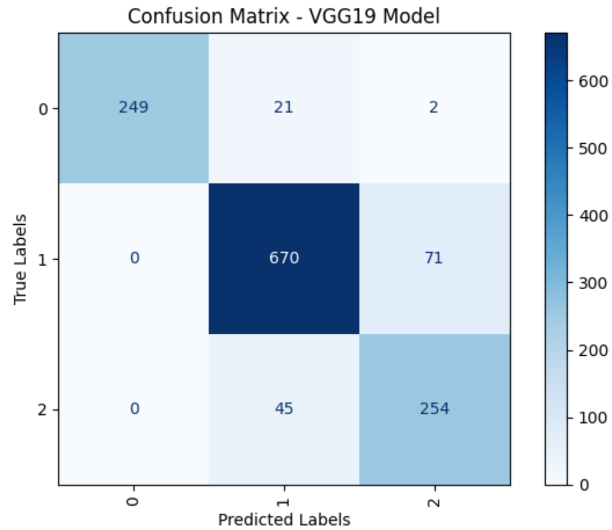

Fig. 7. Confusion Matrix for VGG19

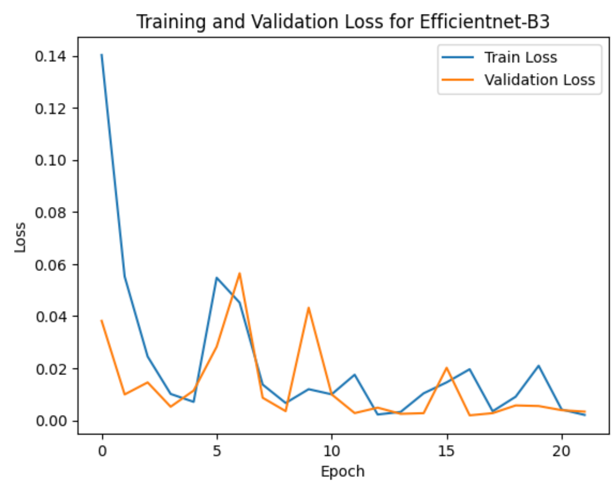

Fig. 8. Loss Curve for EfficientNet_B3 demonstrates a rapid convergence with a steep decline in both training and validation losses, followed by low and erratic values. Both losses are near, indicating consistent training with high generalization, even with little spikes.

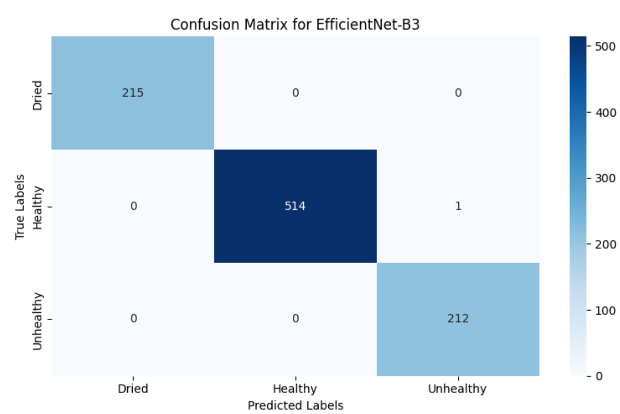

Fig. 9. Confusion Matrix for EfficientNet_B3

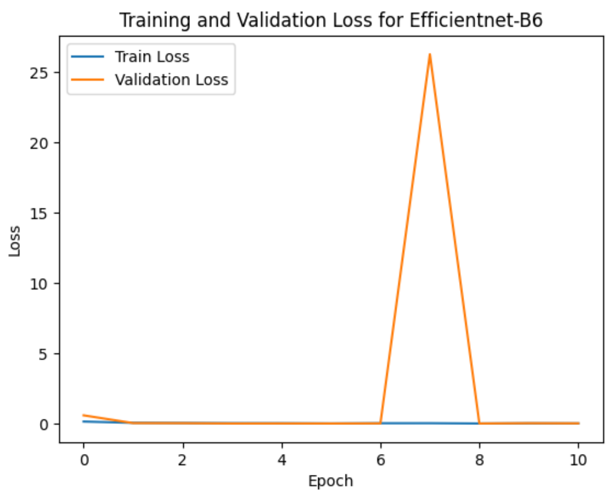

Fig. 10. Loss Curve for EfficientNet_B6 shows that training loss consistently decreased, indicating effective learning. However , the sharp spike in validation loss at epoch 7 indicates a data anomaly during that epoch.

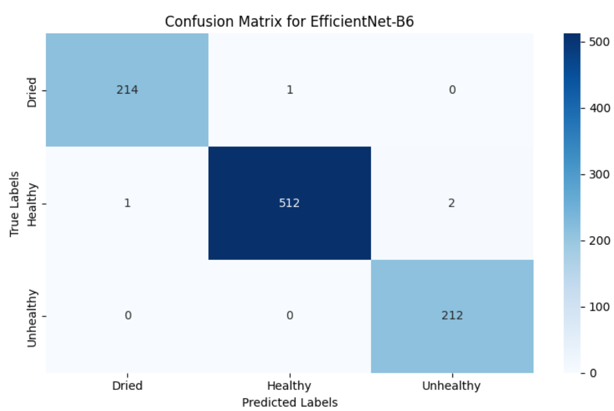

Fig. 11. Confusion Matrix for EfficientNet_B6

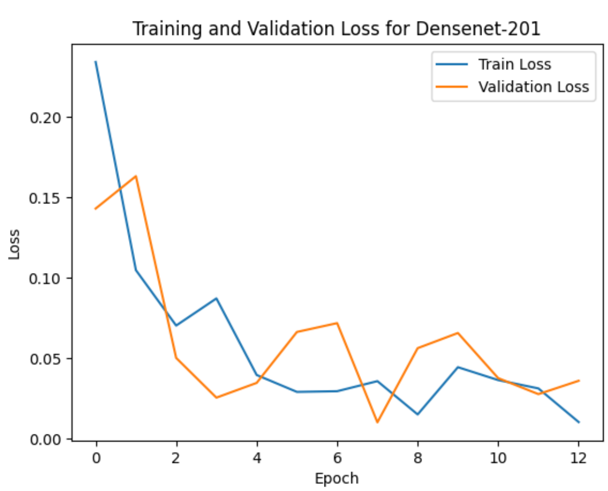

Fig. 12. Loss Curve for DenseNet201 shows both training and validation loss decrease over some epochs, indicating effective learning. The validation loss closely follows the training loss, suggesting minimal overfitting.

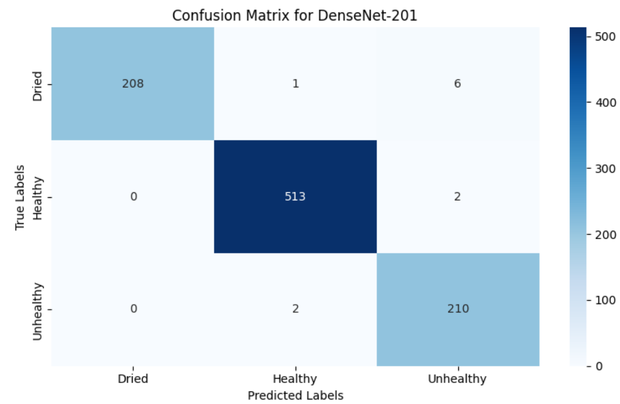

Fig. 13. Confusion Matrix for DenseNet201

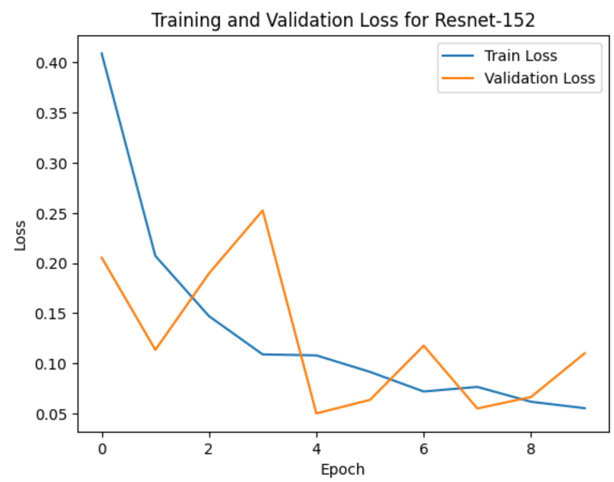

Fig. 14. Loss Curve for ResNet152 shows a steady decrease in training loss and fluctuating but generally declining validation loss. The spikes are due to some data anomaly.

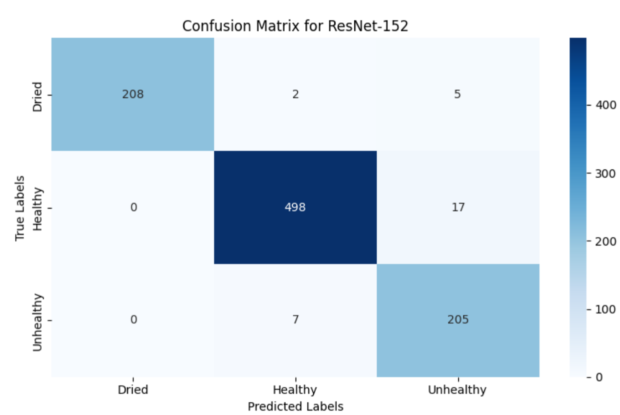

Fig. 15. Confusion Matrix for ResNet152

VII. Grad-CAM Visualization

Grad-CAM uses gradients flowing back from a target class to the last convolutional layer of a model to generate a heatmap. This heatmap highlights the areas in the image that are most relevant to the model's classification for a particular class. Grad-CAM computes the gradients of the target class score concerning the feature maps of a convolutional layer. These gradients are globally pooled to get a single weight for each feature map. Then, each feature map is multiplied by its corresponding weight, and all feature maps are summed up.to the predicted class.

A. XAI (Grad-CAM, Grad-CAM++, Eigen-CAM) and LIME

By applying different XAI methods such as GradCam, GradCam++, Eigen-CAM, and LIME, it was shown that our EfficientNet_B0 model precisely highlighted the important features distinguishing dried,healthy and unhealthy leaves of **Plubmbago Zeylanica** (Chitrak). Hence,validating the interpretability and reliability of our classification results.

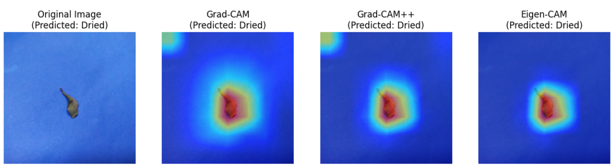

Fig. 16. Image Highlights using XAI.

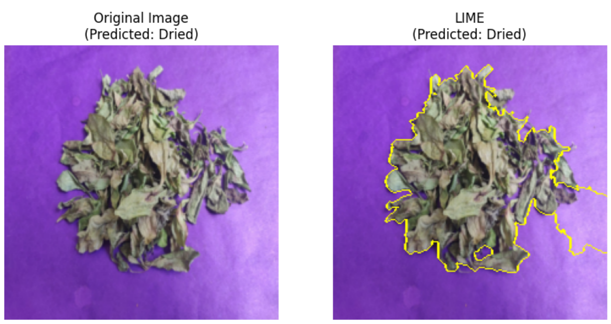

Fig. 17. Image Highlights using LIME.

VIII. Discussion

This study proposed a pipeline for leaf classification. The pipeline is formed of some steps, including a custom CNN, five pre-trained models and XAI generalizations. In this project, we explored the trade-offs between accuracy, computation time and size using models VGG19, ResNet152, EfficientNet_B3 and a custom CNN for leaf classification. The validation accuracies all are above 80%.EfficientNet_B3 provides the highest accuracy.Despite the dataset being balanced, the models consistently performed better on the background class, suggesting it was easier to classify-possibly due to more distinctive visual features. We performed XAI in our dataset, which provided transparency and interpretability to our model. Additionally, we employed LIME(Local Interpretable Model-agnostic Explanations) to interpret model predictions and visualize the decision regions. Some models showed signs of overfitting, and false negatives in diseased leaf detection highlight a need for generalization and possibly more diverse data.

IX. Conclusion

Chitrak is one of the oldest medicinal plants, known for its various properties. This study is on this traditional ayurvedic plant. We successfully classified Plumbago Zeylanica(Chitrak) leaf images into three categories: Healthy, Dired, Unhealthy. For this classification, we implemented multiple deep learning models. The Efficient_B3 performed well with highest accuracy of 99.89%. Other models were also good with high accuracy over 80%.Grad-CAM visualizations were used to interpret model predictions. It helped focusing on the targeted part of the image, ignoring background noise. It provides more accurate predictions by providing visual justification.

Overall, the study's findings demonstrated that using models can produce high accurate classification and identification of medicinal plant leaves.

X. Future Work

For future studies, classification of medicinal plants may include other parts of the plant like flower, stems and barks. Then we may explore others aside from this study, reaching higher accuracy with more accurate prediction. We have also explored some ensemble models, those provide more accurate classification. So, we are also interested in doing so in our future works.

Acknowledgment

We would like to express sincere gratitude to Dr. Mohammad Rifat Ahmmad Rashid, Associate Professor in the Department of Computer Science and Engineering at East West University(EWU), for their invaluable support and guidance throughout this project. His insightful advice, encouragement, and expertise have been crucial in the successful completion of the work.

References

[1] Dhoka, Amit; Suryawanshi, Yogesh; Patil, Kailas (2023), "Plumbago Zeylanica (Chitrak) Leaf Image Dataset," Mendeley Data, V2. Available: https://doi.org/10.17632/twpv7hhmgb.2

[2] N. Althuniyan and A. R. Al-Shamasneh, "DeepLeaf: Automated Leaves Classification Using Convolutional Neural Networks," *European Scientific Journal (ESJ)*, vol. 31, p. 303, 2024. Available: https://eujournal.org/index.php/esj/article/view/18343

[3] T. Siddharth, B. S. Kirar, and D. K. Agrawal, "Plant Species Classification Using Transfer Learning by Pre Trained Classifier VGG-19," *arXiv preprint*, arXiv:2209.03076, 2022. Available: https://arxiv.org/abs/2209.03076

[4] O. Attallah, "Tomato Leaf Disease Classification via Compact Convolutional Neural Networks with Transfer Learning and Feature Selection," *Horticulturae*, vol. 9, no. 2, p. 149, 2023. Available: https://www.mdpi.com/2311-7524/9/2/149

[5] C.-P. Lee, K. M. Lim, Y. X. Song, and A. Alqahtani, "Plant-CNN-ViT: Plant Classification with Ensemble of Convolutional Neural Networks and Vision Transformer," *Plants*, vol. 12, no. 14, p. 2642, 2023. Available: https://www.mdpi.com/2223-7747/12/14/2642

[6] K. Gopalan, S. Srinivasan, P. Pragya, M. Singh, S. K. Mathivanan, and U. Moorthy, "Corn Leaf Disease Diagnosis: Enhancing Accuracy with ResNet152 and Grad-CAM for Explainable AI," *BMC Plant Biology*, 2025.
Available: https://bmcplantbiol.biomedcentral.com/articles/10.1186/s12870-025-06386-0

[7] K. U. K. Rodrigo, J. H. H. S. Marcial, S. C. Brillo, K. E. Mata, and J. C. Morano, "An Enhancement of CNN Algorithm for Rice Leaf Disease Image Classification in Mobile Applications," *Journal of Information Systems Engineering and Management*, vol. 10, no. 6s, 2024.
Available: https://jisem-journal.com/index.php/journal/article/view/720

[8] E. F. Custodio, "Improved medicinal plant leaf classification using transfer learning and model averaging ensemble of deep convolutional neural networks," in *Proceedings of the 2024 International Conference on Information Technology Research and Innovation (ICITRI)*, Sep. 2024.
Available:https://www.researchgate.net/publication/384585928_Improved_Medicinal_Plant_Leaf_Classification_using_Transfer_Learning_and_Model_Averaging_Ensemble_of_Deep_Convolutional_Neural_Networks

[9] U. Schippmann, J. Danna, J. Leaman, and A. B. Cunningham (2022). "Impact of Cultivation and Gathering of Medicinal Plants on Biodiversity: Global Trends and Issues."
Available: https://www.researchgate.net/publication/265157471_Impact_of_Cultivation_and_Gathering_of_Medicinal_Plants_on_Biodiversity_Global_Trends_and_Issues/citations

[10] A. K. Mulugeta, D. P. Sharma, and A. H. Mesfin (2024). "Deep learning for medicinal plant species classification and recognition: a systematic review," *Frontiers in Plant Science*.
Available: https://www.frontiersin.org/journals/plant-science/articles/10.3389/fpls.2023.1286088/full

[11] H. Bouakkaz, M. Bouakkaz, C. A. Kerrache, and S. Dhelim (2025). "Enhanced classification of medicinal plants using deep learning and optimized CNN architectures," *Heliyon*.
Available: https://doi.org/10.1016/j.heliyon.2025.e42385