AUTOMATED MEDICINAL PLANT IDENTIFICATION

Minor Project (PR - 591)

Project Submitted in Partial Fulfillment of the Requirements for the Degree of Bachelor of Technology in the field of Computer Science and Engineering

BY
Mainak Ghosh (123211003072)
Ishani Mitra (123211003067)
Mamartha Dutta (123211003074)
Pritha Gupta (123211003099)
Diptanu Das (123211003059)

Under the supervision
of
Ms. Debasree Mitra

Department of Computer Science and Engineering
JIS College of Engineering

Block-A, Phase-III, Kalyani, Nadia, Pin-741235
West Bengal, India
Dec, 2023

JIS College of Engineering
Block 'A', Phase-III, Kalyani, Nadia, 741235
Phone: +91 33 2582 2137, Telefax: +91 33 2582 2138
Website: www.jiscollege.ac.in, Email: info@jiscollege.ac.in

CERTIFICATE

This is to certify that **Mainak Ghosh (123211003072), Ishani Mitra (123211003067), Mamartha Dutta (123211003074), Pritha Gupta (123211003099**), **Diptanu Das (123211003059**) have completed their project entitled **Automated Medicinal Plant Identification,** under the guidance of **Ms. Debasree Mitra** in partial fulfillment of the requirements for the award of the **Bachelor of Technology in Computer Science and Engineering** from JIS college of Engineering (An Autonomous Institute) is an authentic record of their own work carried out during the academic year 2023-24 and to the best of our knowledge, this work has not been submitted elsewhere as part of the process of obtaining a degree, diploma, fellowship or any other similar title.

---------------------------------- ------------------------------

Signature of the Supervisor **Signature of the HOD**

Place:

Date:

ACKNOWLEDGEMENT

The analysis of the project work wishes to express our gratitude to Ms. Debasree Mitra for allowing the degree attitude and providing effective guidance in development of this project work. Her conscription of the topic and all the helpful hints, she provided, contributed greatly to successful development of this work, without being pedagogic and overbearing influence.

We also express our sincere gratitude to Dr. Bikramjit Sarkar, Head of the Department of Computer Science and Engineering of JIS College of Engineering and all the respected faculty members of Department of CSE for giving the scope of successfully carrying out the project work.

Finally, we take this opportunity to thank to Prof. **(Dr.) Partha Sarkar**, Principal of JIS College of Engineering for giving us the scope of carrying out the project work.

Date:

……………………………………………………………………

MAINAK GHOSH
B. TECH in Computer Science and Engineering
3^{rd} YEAR/5^{th} SEMESTER
Univ Roll--123211003072

……………………………………………………………………

ISHANI MITRA
B. TECH in Computer Science and Engineering
3^{rd} YEAR/5^{th} SEMESTER
Univ Roll—123211003067

……………………………………………………………………

MAMARTHA DUTTA
B. TECH in Computer Science and Engineering
3^{rd} YEAR/5^{th} SEMESTER
Univ Roll—123211003074

……………………………………………………………………

PRITHA GUPTA
B. TECH in Computer Science and Engineering
3^{rd} YEAR/5^{th} SEMESTER
Univ Roll--123211003099

……………………………………………………………………

DIPTANU DAS
B. TECH in Computer Science and Engineering
3^{rd} YEAR/5^{th} SEMESTER Univ Roll—123211003059

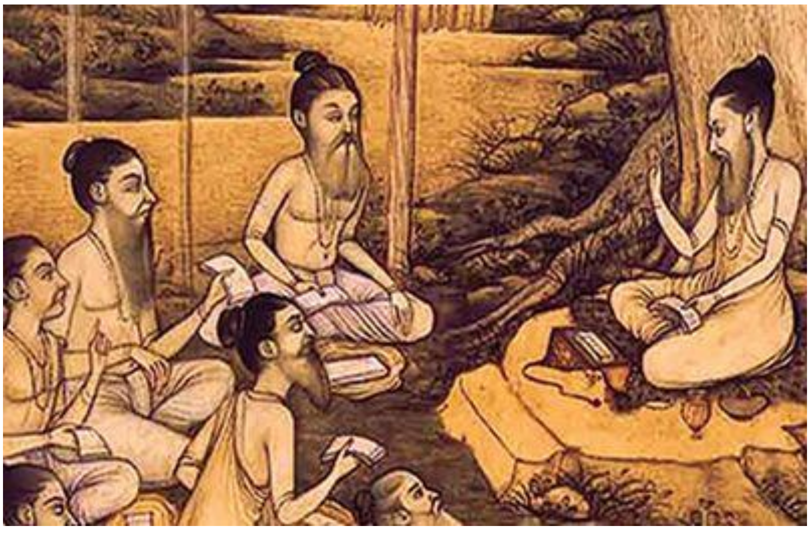
CONTENTS

ABSTRACT

The proper identification of plant species has major benefits for a wide range of stakeholders ranging from forestry services, botanists, taxonomists, physicians, pharmaceutical laboratories, organizations fighting for endangered species, government and the public at large. Consequently, this has fueled an interest in developing automated systems for the recognition of different plant species. A fully automated method for the recognition of medicinal plants using computer vision and machine learning techniques has been presented. In this project we explore feature vectors from both the front and back side of a green leaf along with morphological features to arrive at a
unique optimum combination of features that maximizes the identification rate. A database of medicinal plant leaves is created from scanned images of front and back side of leaves of commonly used medicinal plants. Leaves from different medicinal plant species were collected and photographed using a webcam in a laboratory setting. A large number of features were extracted from each leaf such as its length, width, perimeter, area, number of vertices, color, perimeter and area of hull. Several derived features were then computed from these attributes. The best results were obtained from a random forest classifier using a 10-fold cross-validation technique. With an accuracy of 90.1%, the random forest classifier performed better than other machine learning approaches such as the k-nearest neighbor, naïve Bayes, support vector machines and neural networks. These results are very encouraging and future work will be geared towards using a larger dataset and high-performance computing facilities to investigate the performance of deep learning neural networks to identify medicinal plants used in primary health care. To the best of our knowledge, this work is the first of its kind to have created a unique image dataset for medicinal plants. It is anticipated that a web-based or mobile computer system for the automatic recognition of medicinal plants will help the local population to improve their knowledge on medicinal plants, help taxonomists to develop more efficient species identification techniques and will also contribute significantly in the protection of endangered species.

INTRODUCTION

The world bears thousands of plant species, many of which have medicinal values, others are close to extinction, and still others that are harmful to man. Not only are plants an essential resource for human beings, but they form the base of all food chains. The medicinal plants are used mostly in herbal, ayurvedic and folk medicinal manufacturing. Herbal plants are plants that can be used for alternatives to cure diseases naturally. About 80% of people in the world still depend on traditional medicine. Meanwhile, according to herbal plants are plants whose plant parts (leaves, stems, or roots) have properties that can be used as raw materials in making modern medicines or traditional medicines. These medicinal plants are often found in the forest. There are various types of herbal plants that we can know through the identification of these herbs, one of which is using identification through the leaves. and protect plant species, it is crucial to study and classify plants correctly. Combinations of a small subset amounting to 1500 of these plants are used in Herbal medicines of different systems of India. Specifically, commercial Ayurvedic preparations use 500 of these plants. Over 80% of plants used in ayurvedic formulations are collected from the forests and wastelands whereas the remaining are cultivated in agricultural lands. More than 8000 plants of Indian origin have been found to be of medicinal value.



MOTIVATION

In the ancient past, the Ayurvedic physicians themselves picked the medicinal plants and prepared the medicines for their patients. Today only a few practitioners follow this practice. The manufacturing and marketing of Ayurvedic drugs has become a thriving industry whose turnover exceeds Rs. 4000 crores. The number of licensed Ayurvedic medicine manufacturers in India easily exceeds 8500. This commercialization of Ayurvedic sector has brought in to focus several questions regarding the quality of raw materials used for Ayurvedic medicines. Today the plants are collected by women and children from forest areas; those are not professionally trained in identifying correct medicinal plants. Manufacturing units often receive incorrect or substituted medicinal plants. Most of these units lack adequate quality control mechanisms to screen these plants. In addition to this, confusion due to variations in local name is also rampant. Some plants arrive in dried form and this make the manual identification task much more difficult. Incorrect use of medicinal plants makes the Ayurvedic medicine ineffective. It may produce unpredictable side effects also. In this situation, strict measures for quality control must be enforced on Ayurvedic medicines and raw materials used by the industry in order to sustain the present growth of industry by maintaining the efficacy and credibility of medicines. A trained Botanist looks for all the available features of the plants such as leaves, flowers, seeds, root and stem to identify plants. Except for the leaf, all others are 3D objects and increase the complexity of analysis by computer. However, plant leaves are 2D objects and carry sufficient information to identify the plant. Leaves can be collected easily and image acquisition may be carried out using inexpensive digital cameras, mobile phones or document scanners. It is available at any time of the year in contrast to flowers and seeds. Leaves acquire a specific color, texture and shape when it grows and these changes are relatively insignificant. Plant recognition based on leaves depends on finding exact descriptors and extracting the feature vectors from it. Then the feature vectors of the training samples are compared with the feature vectors of the test sample to find the degree of similarity using an appropriate classifier.

PROBLEM DEFINITION

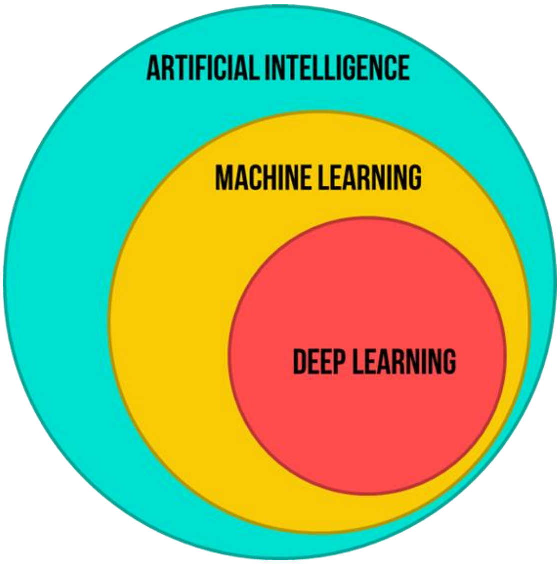
Deep learning is one of the major subfields of machine learning framework. Machine learning is the study of design of algorithms, inspired from the model of human brain. Deep learning is becoming more popular in data science fields like robotics, artificial intelligence (AI), audio & video recognition and image recognition. Artificial neural network is the core of deep learning methodologies. Deep learning is supported by various libraries such as Theano, TensorFlow, Caffe, Mx net etc., Keras is one of the most powerful and easy to use python library, which is built on top of popular deep learning libraries like TensorFlow, Theano, etc., for creating deep learning models. Detection of correct medicinal leaves can help botanists, taxonomists and drug manufacturers to make quality drug and can reduce the side effects caused by the wrong drug delivery. To identify the leaves of the plants, a type of artificial neural network called convolutional neural network (CNN) is used. The architecture we used here is Densenet121, which is a convolutional neural network that is a powerful model capable of achieving high accuracies on challenging datasets. To address these challenges, there is a pressing need for an automated system that can accurately and efficiently identify medicinal plants based on their visual characteristics. Leveraging advancements in deep learning and machine learning techniques, this study aims to develop a robust and scalable solution for automated medicinal plant identification. The primary objective is to create a model that can analyze plant images, extract discriminative features, and classify medicinal plant species accurately and rapidly, thereby mitigating the limitations of manual identification methods.

By employing a combination of deep convolutional neural networks (CNNs) for feature extraction and machine learning algorithms for classification, this research seeks to create a hybrid system capable of revolutionizing the identification process. This system aims to provide a reliable and time-efficient means of identifying medicinal plants, contributing significantly to the fields of healthcare, pharmaceuticals, and biodiversity conservation.

LITERATURE SURVEY

- **ARTIFICIAL INTELLIGENCE:**

According to the father of Artificial Intelligence, John McCarthy, it is “The science and engineering of making intelligent machines, especially intelligent computer programs”.



Artificial Intelligence is a way of making a computer, a computer-controlled robot, or a software think intelligently, in the similar manner the intelligent humans think.

AI is accomplished by studying how human brain thinks, and how humans learn, decide, and work while trying to solve a problem, and then using the outcomes of this study as a basis of developing intelligent software and systems.

The development of AI started with the intention of creating similar intelligence in machines that we find and regard high in humans. To Create Expert Systems − The systems which exhibit intelligent behavior, learn, demonstrate, explain, and advice its users.
To Implement Human Intelligence in Machines − Creating systems that understand, think, learn, and behave like humans.

Applications of AI

AI has been dominant in various fields such as: -

Gaming − AI plays crucial role in strategic games such as chess, poker, tic-tac-toe, etc., where machine can think of large number of possible positions based on heuristic knowledge.

Natural Language Processing − It is possible to interact with the computer that understands natural language spoken by humans.

Expert Systems − There are some applications which integrate machine, software, and special information to impart reasoning and advising. They provide explanation and advice to the users.

Vision Systems − These systems understand, interpret, and comprehend visual input on the computer.

For example: A spying aeroplane takes photographs, which are used to figure out spatial information Or map of the areas. Doctors use clinical expert system to diagnose the patient. Police use computer software that can recognize the face of criminal with the stored portrait made by forensic artist.

Speech Recognition − Some intelligent systems are capable of hearing and comprehending the language in terms of sentences and their meanings while a human talks to it. It can handle different accents, slang words, noise in the background, change in human's noise due to cold, etc.

Handwriting Recognition − The handwriting recognition software reads the text written on paper by a pen or on screen by a stylus. It can recognize the shapes of the letters and convert it into editable text.

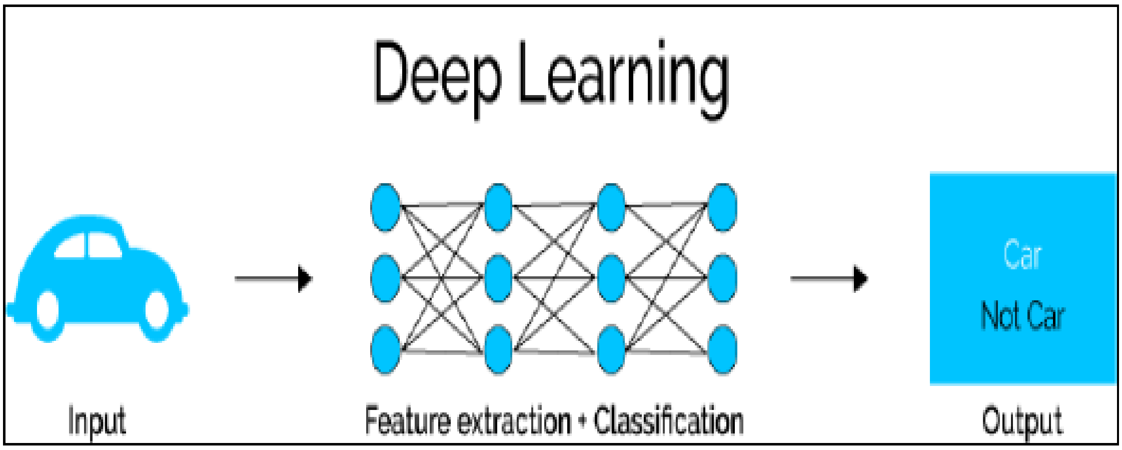
Intelligent Robots − Robots are able to perform the tasks given by a human. They have sensors to detect physical data from the real world such as light, heat, temperature, movement, sound, bump, and pressure. They have efficient processors, multiple sensors and huge memory, to exhibit intelligence. In addition, they are capable of learning from their mistakes and they can adapt to the new environment.

- **<u>DEEP LEARNING</u>**

Deep learning is a subset of machine learning. Usually, when people use the term deep learning, they are referring to deep artificial neural networks, and somewhat less frequently to deep reinforcement learning.

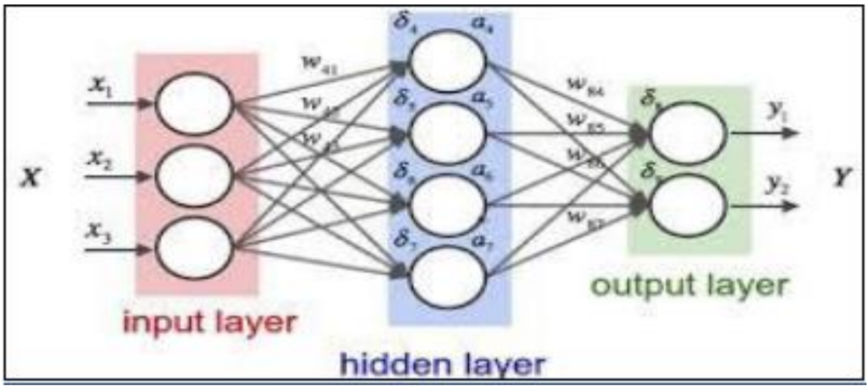
Deep learning is a class of machine learning algorithms that:

- Use a cascade of multiple layers of nonlinear processing units for feature extraction and transformation. Each successive layer uses the output from the previous layer as input.
- Learn in supervised (e.g., classification) and/or unsupervised (e.g., pattern analysis) manners.
- Learn multiple levels of representations that correspond to different levels of abstraction; the levels form a hierarchy of concepts.
- Use some form of gradient descent for training via backpropagation.



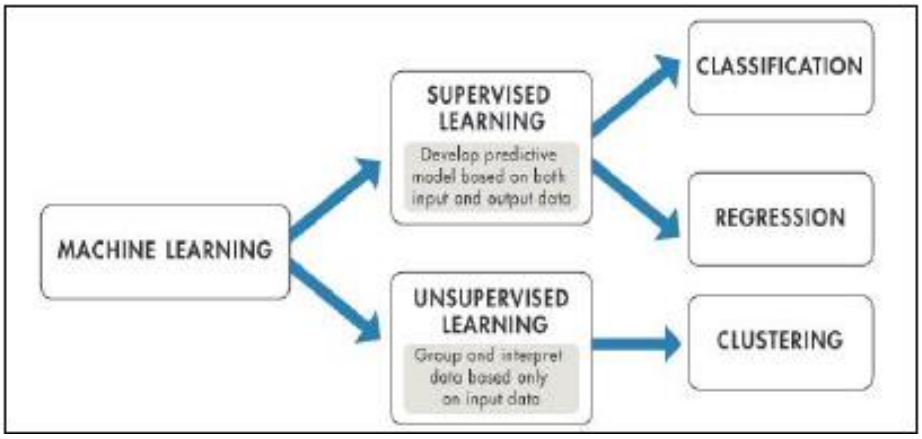
- **NEURAL NETWORKING**

Artificial neural networks (ANNs) or **connectionist systems** are computing systems inspired by the biological neural networks that constitute animal brains. Such systems learn (progressively improve performance on) tasks by considering examples, generally without task-specific programming



An ANN is based on a collection of connected units or nodes called artificial neurons (analogous to biological neurons in an animal brain). Each connection between artificial neurons can transmit a signal from one to another.

- **MACHINE LEARNING**



Machine learning is a field of computer science that gives computers the ability to learn without being explicitly programmed.

Evolved from the study of pattern recognition and computational learning theory in artificial intelligence, machine learning explores the study and construction of algorithms that can learn from and make predictions on data.

- Literature Survey on Automated Medicinal Plants Identification Using Deep Learning and Machine Learning Algorithms

1. Overview of Traditional Methods:

- Reviewing conventional methods of medicinal plant identification, including botanical expertise, dichotomous keys, and morphological characteristics assessment.

2. Evolution of Computer Vision in Plant Identification:

Exploring the evolution of computer vision techniques applied to plant identification, highlighting advancements in image processing, feature extraction, and classification algorithms.

3. Deep Learning Applications in Plant Identification:

Reviewing studies utilizing deep learning techniques, particularly convolutional neural networks (CNNs), for plant species identification.

4. Machine Learning Algorithms for Plant Classification:

Discussing the use of various machine learning algorithms (e.g., Random Forest, Support Vector Machines, k-Nearest Neighbors) in plant species classification based on visual features.

5. Datasets and Image Preprocessing Techniques:

Surveying existing datasets of medicinal plant images and examining methodologies for image preprocessing, normalization, and augmentation for improved model performance.

6. Hybrid Approaches and Ensemble Methods:

Investigating studies that combine deep learning architectures with traditional machine learning algorithms to create hybrid models for more accurate and robust plant identification.

7. Challenges and Limitations:

Discussing the challenges faced in automated plant identification, such as dataset scarcity, class imbalance, domain adaptation, and the need for interpretability in models.

8. Applications and Future Directions:

Examining real-world applications of automated medicinal plant identification in healthcare, drug discovery, and conservation efforts.

9. Comparison and Evaluation Metrics:

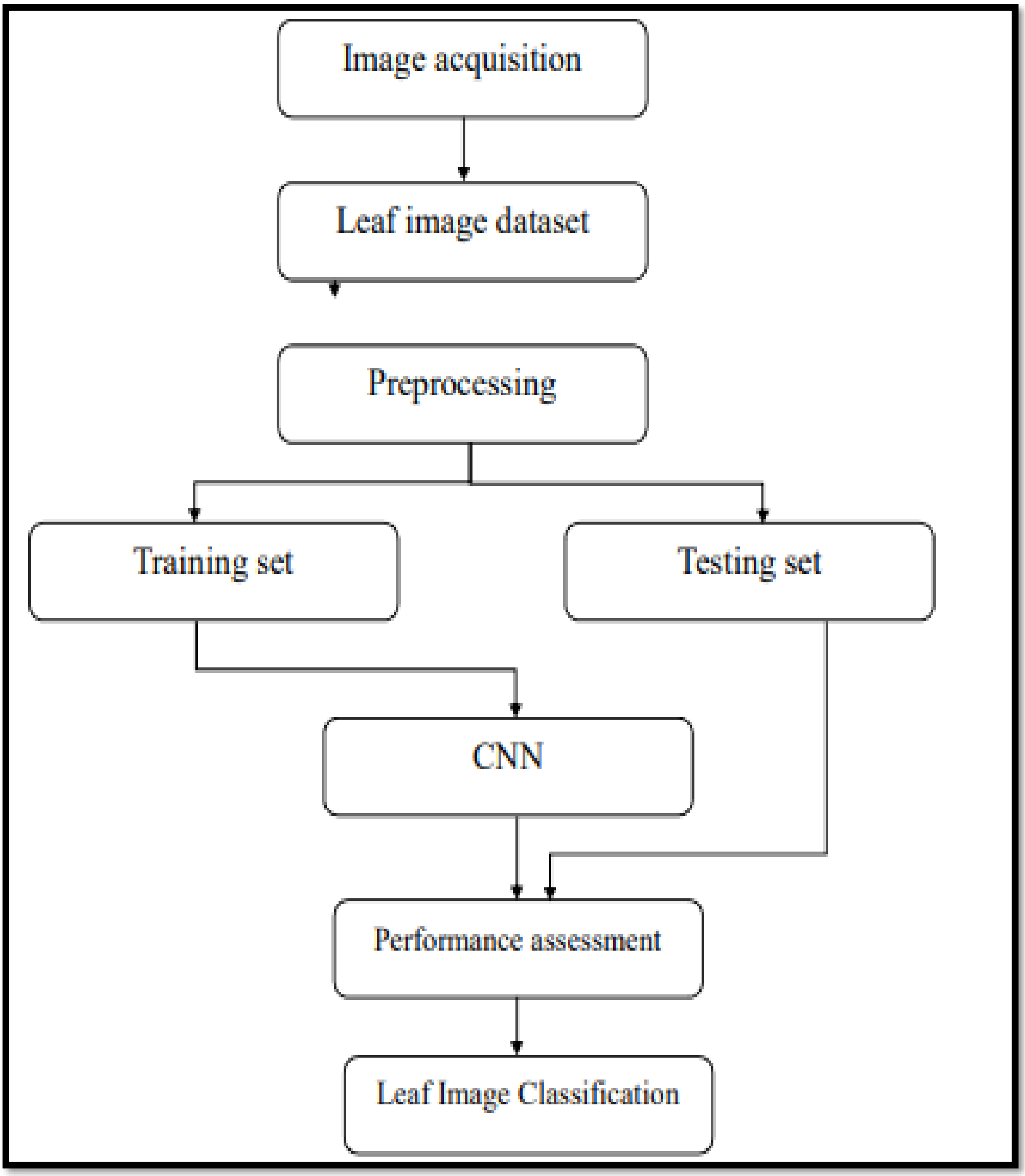
Analyzing various evaluation metrics used in assessing the performance of automated plant identification models, such as accuracy, precision, recall, and F1-score.

10. Current State-of-the-Art and Gaps in Research:

Summarizing the current state-of-the-art methodologies, identifying gaps in research, and suggesting potential directions for future studies.

This literature survey aims to provide a comprehensive overview of the existing research landscape concerning automated medicinal plants identification using deep learning and machine learning algorithms. It covers various aspects, from traditional methods to state-of-the-art techniques, highlighting advancements, challenges, and future prospects in this domain.

Flow Chart for Program's Methodology



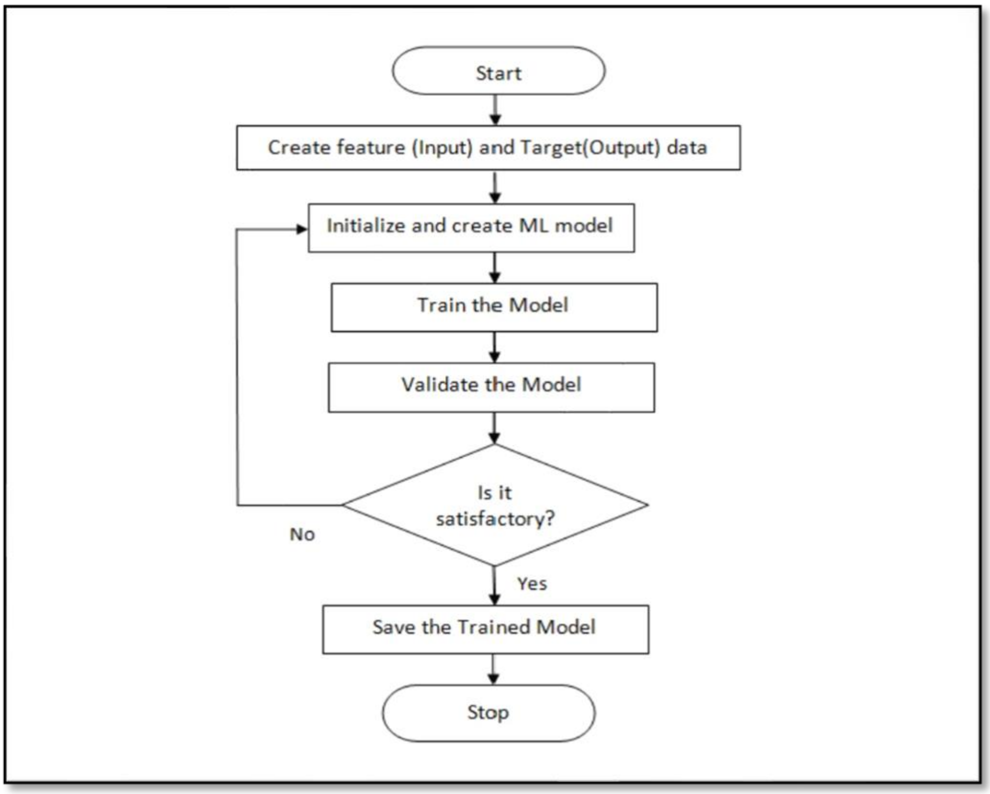


Figure 1: Training of Model

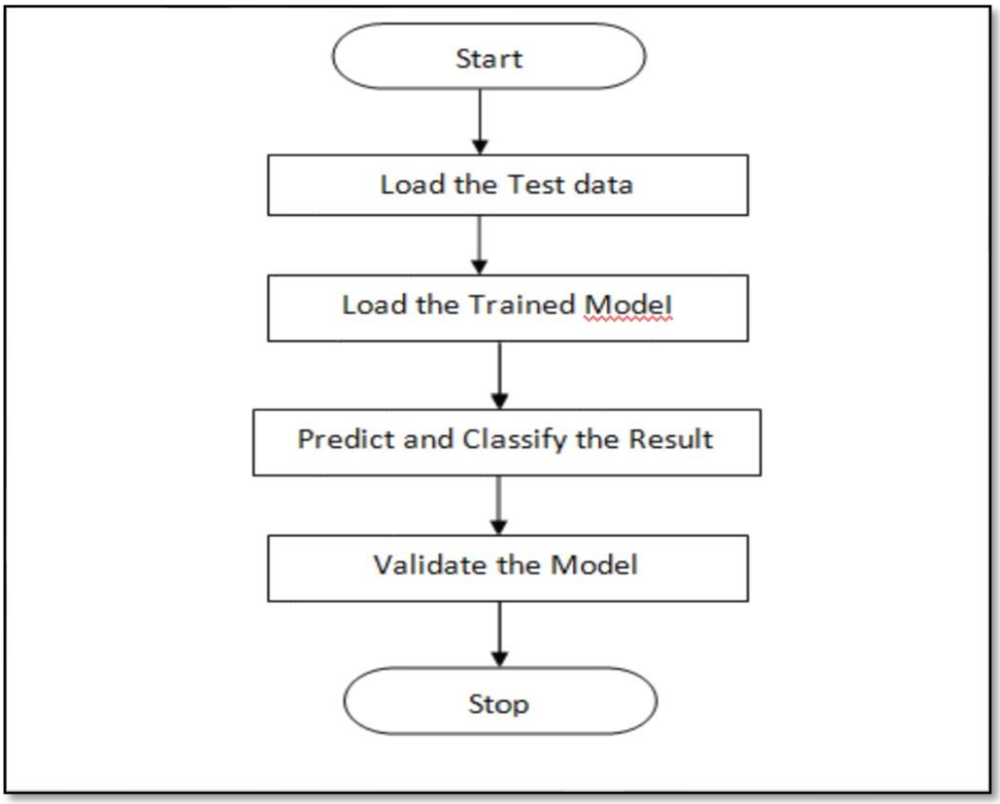


Figure 2: Testing of Model

METHODOLOGY

1. Data Collection:

Dataset Acquisition: Acquired a diverse dataset containing images of various medicinal plants, encompassing different species and parts of the plants (leaves, flowers, roots).

Data Annotation: Manually labeled the dataset with accurate plant species or medicinal part Information for supervised learning.

Figure 1: Dataset of Different Medicinal Plants

2. Data Preprocessing:

Data Cleaning: Removed duplicates, irrelevant images, and performed augmentation techniques (resizing, cropping, rotations) to enhance dataset variability and balance classes.

Image Processing: Standardized images by resizing and normalizing to optimize model performance.

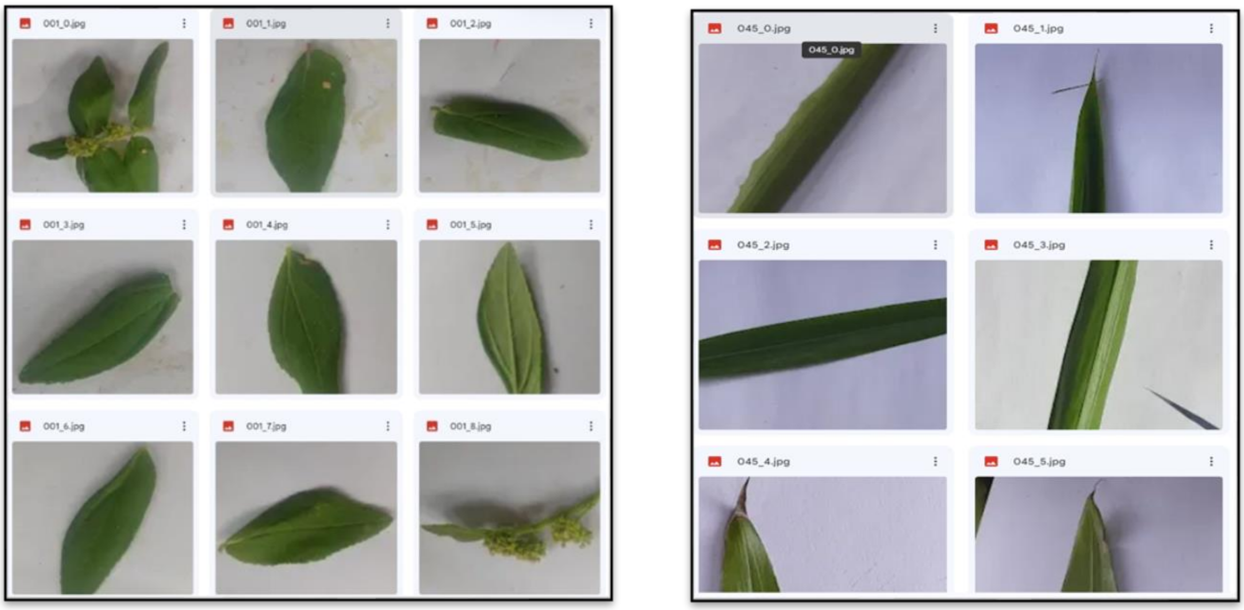


Figure 2: Standardized images

3. Model Development by VGG16:

Feature Extraction: Utilized transfer learning with pre-trained VGG-16 CNN architecture to extract relevant features from the images.

```python
# Define the image dimensions and batch size
img_height, img_width = 224, 224
batch_size = 32

# Load the pre-trained VGG16 model (without the top layers)
base_model = VGG16(weights='imagenet', include_top=False, input_shape=(img_height, img_width, 3))

# Freeze the pre-trained layers so they are not trainable
for layer in base_model.layers:
    layer.trainable = False

# Define the number of classes in your dataset
num_classes = 50  # Change this to the actual number of classes in your dataset

# Build the model
model = Sequential([
    base_model,
    Flatten(),
    Dense(256, activation='relu'),
    Dense(128, activation='relu'),
    Dense(num_classes, activation='softmax')  # Use the actual number of classes
])
```

Figure 3: Using VGG-16 for training the model

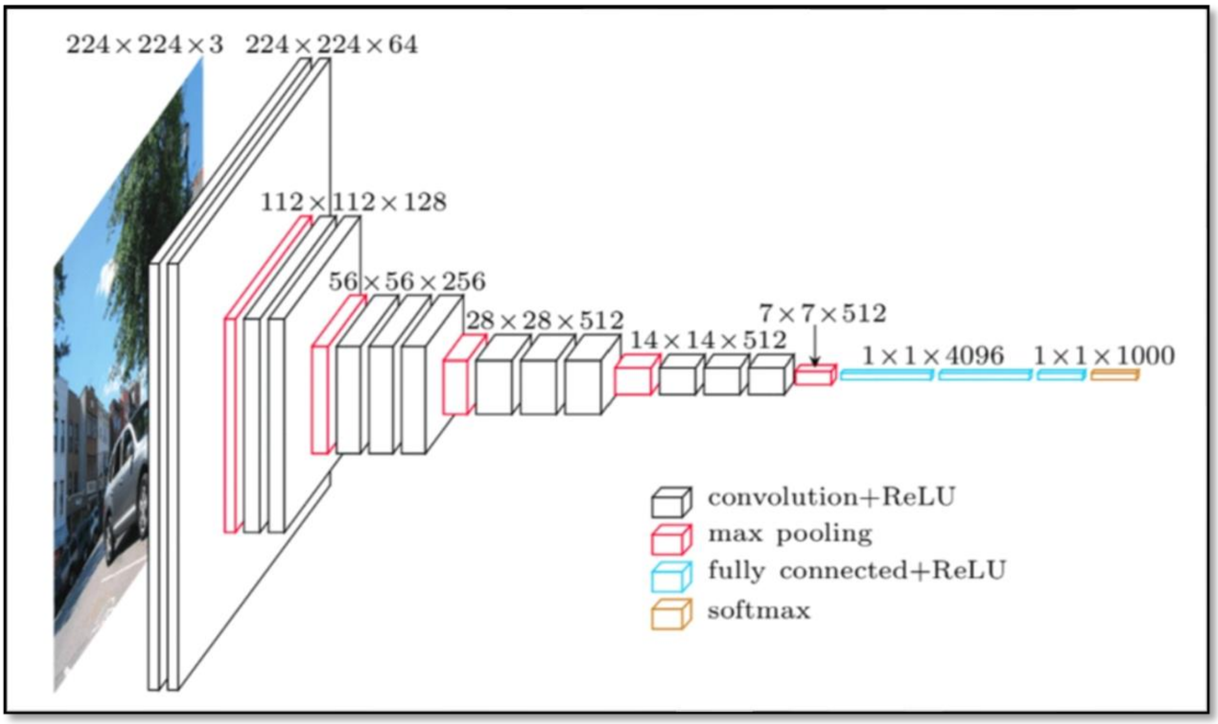


Figure 6: VGG-16 Architecture

<u>Model Selection and Training</u>: Explored various architectures and hyperparameters, selecting the best-performing model through a train-validation split. Trained the model on the training set and optimized using the validation set.

```python
# Train the model with early stopping
epochs = 25
history = model.fit(
    train_generator,
    epochs=epochs,
    validation_data=validation_generator #,
    #callbacks=[early_stopping]   Add the early stopping callback here
)
```

Figure 7: Training the Model

```
Found 3047 images belonging to 50 classes.
Found 738 images belonging to 50 classes.
Epoch 1/25
96/96 [==============================] - 91s 922ms/step - loss: 2.9104 - accuracy: 0.2708 - val_loss: 1.9676 - val_accuracy: 0.4783
Epoch 2/25
96/96 [==============================] - 86s 897ms/step - loss: 1.3445 - accuracy: 0.6154 - val_loss: 1.1917 - val_accuracy: 0.6680
Epoch 3/25
96/96 [==============================] - 88s 918ms/step - loss: 0.7981 - accuracy: 0.7719 - val_loss: 0.9972 - val_accuracy: 0.7168
Epoch 4/25
96/96 [==============================] - 87s 907ms/step - loss: 0.5617 - accuracy: 0.8385 - val_loss: 0.9963 - val_accuracy: 0.6965
Epoch 5/25
96/96 [==============================] - 91s 952ms/step - loss: 0.4334 - accuracy: 0.8687 - val_loss: 0.9346 - val_accuracy: 0.7222
Epoch 6/25
96/96 [==============================] - 88s 921ms/step - loss: 0.3659 - accuracy: 0.8947 - val_loss: 0.8229 - val_accuracy: 0.7602
Epoch 7/25
96/96 [==============================] - 88s 916ms/step - loss: 0.3098 - accuracy: 0.9111 - val_loss: 0.7953 - val_accuracy: 0.7656
Epoch 8/25
96/96 [==============================] - 88s 918ms/step - loss: 0.2414 - accuracy: 0.9265 - val_loss: 0.8066 - val_accuracy: 0.7602
Epoch 9/25
96/96 [==============================] - 88s 914ms/step - loss: 0.1911 - accuracy: 0.9393 - val_loss: 0.7757 - val_accuracy: 0.8049
Epoch 10/25
96/96 [==============================] - 91s 952ms/step - loss: 0.1997 - accuracy: 0.9383 - val_loss: 0.7111 - val_accuracy: 0.8008
Epoch 11/25
96/96 [==============================] - 87s 910ms/step - loss: 0.1567 - accuracy: 0.9550 - val_loss: 0.8597 - val_accuracy: 0.7696
Epoch 12/25
96/96 [==============================] - 89s 927ms/step - loss: 0.1300 - accuracy: 0.9554 - val_loss: 0.6769 - val_accuracy: 0.8184
Epoch 13/25
96/96 [==============================] - 88s 922ms/step - loss: 0.1128 - accuracy: 0.9662 - val_loss: 0.7401 - val_accuracy: 0.8211
Epoch 14/25
96/96 [==============================] - 88s 920ms/step - loss: 0.0946 - accuracy: 0.9764 - val_loss: 0.6727 - val_accuracy: 0.8089
Epoch 15/25
96/96 [==============================] - 89s 927ms/step - loss: 0.0820 - accuracy: 0.9770 - val_loss: 0.7781 - val_accuracy: 0.8144
Epoch 16/25
96/96 [==============================] - 89s 923ms/step - loss: 0.0779 - accuracy: 0.9774 - val_loss: 0.7047 - val_accuracy: 0.8279
Epoch 17/25
96/96 [==============================] - 88s 919ms/step - loss: 0.0816 - accuracy: 0.9737 - val_loss: 0.6636 - val_accuracy: 0.8306
Epoch 18/25
96/96 [==============================] - 92s 962ms/step - loss: 0.0571 - accuracy: 0.9849 - val_loss: 1.0807 - val_accuracy: 0.7561
Epoch 19/25
96/96 [==============================] - 89s 930ms/step - loss: 0.1195 - accuracy: 0.9603 - val_loss: 0.8657 - val_accuracy: 0.7886
Epoch 20/25
96/96 [==============================] - 90s 939ms/step - loss: 0.1294 - accuracy: 0.9570 - val_loss: 1.1044 - val_accuracy: 0.7547
Epoch 21/25
96/96 [==============================] - 89s 924ms/step - loss: 0.0859 - accuracy: 0.9698 - val_loss: 1.1472 - val_accuracy: 0.7398
Epoch 22/25
96/96 [==============================] - 92s 960ms/step - loss: 0.0706 - accuracy: 0.9764 - val_loss: 0.7993 - val_accuracy: 0.8306
Epoch 23/25
96/96 [==============================] - 88s 918ms/step - loss: 0.0713 - accuracy: 0.9777 - val_loss: 0.7357 - val_accuracy: 0.8293
Epoch 24/25
96/96 [==============================] - 89s 928ms/step - loss: 0.0784 - accuracy: 0.9731 - val_loss: 1.1047 - val_accuracy: 0.7710
Epoch 25/25
96/96 [==============================] - 90s 940ms/step - loss: 0.1309 - accuracy: 0.9541 - val_loss: 1.8143 - val_accuracy: 0.6518
```

Figure 8: Epochs

4. Model Validation and Testing by VGG16:

Evaluation Metrics: Utilized metrics such as accuracy, precision, recall, and F1-score to assess the model's performance.

Test Set Evaluation: Validated the model's performance on a separate test set to ensure its generalizability.

```python
# Plot training accuracy and validation accuracy over epochs
import matplotlib.pyplot as plt

plt.figure(figsize=(8, 5))
plt.plot(history.history['accuracy'], label='Training Accuracy', marker='o')
plt.plot(history.history['val_accuracy'], label='Validation Accuracy', marker='o')
plt.title('Training and Validation Accuracy')
plt.xlabel('Epoch')
plt.ylabel('Accuracy')
plt.legend()
plt.grid(True)
plt.show()
```

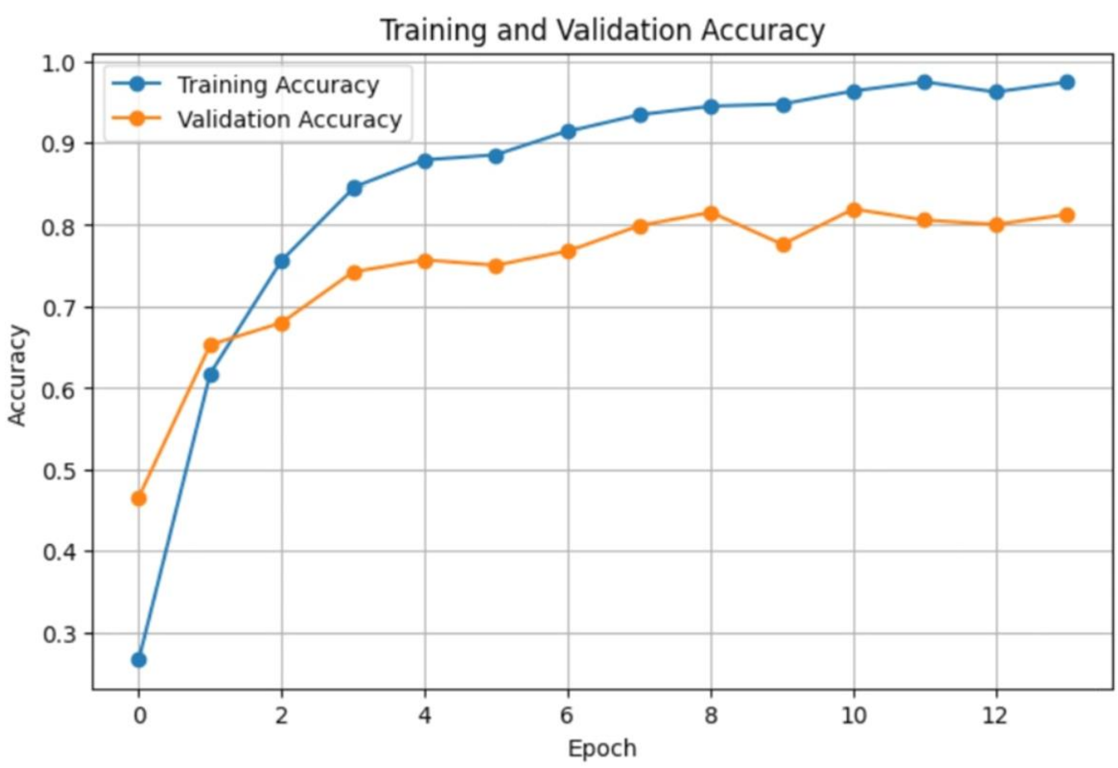


Figure 9: Graph of Training and Validation Accuracy over Epoch

5. Model Deployment By CNN:

Feature Extraction: Utilized transfer learning with pre-trained CNN architecture to extract relevant features from the images.

```python
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense, Conv2D, MaxPooling2D, Flatten

# Build the CNN model
model = Sequential([
    Conv2D(32, (3, 3), activation='relu', input_shape=(img_height, img_width, 3)),
    MaxPooling2D(pool_size=(2, 2)),
    Conv2D(64, (3, 3), activation='relu'),
    MaxPooling2D(pool_size=(2, 2)),
    Flatten(),
    Dense(64, activation='relu'),
    Dense(num_classes, activation='softmax')
])

# Compile the model
model.compile(optimizer='adam', loss='categorical_crossentropy', metrics=['accuracy'])

# Train the model
epochs = 5  # Adjust the number of epochs as needed
model.fit(
    train_generator,
    epochs=epochs,
    validation_data=validation_generator
)
```

Figure 10: Using CNN for training the model

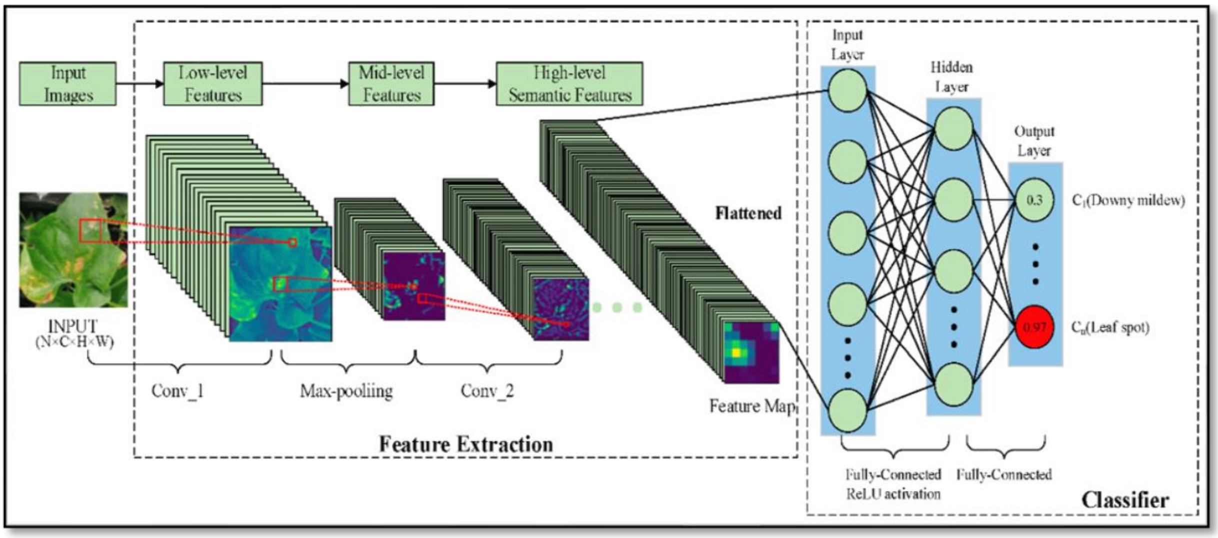


Figure 11: CNN Architecture.

Model Selection and Training: Explored various architectures and hyperparameters, selecting the best-performing model through a train-validation split. Trained the model on the training set and optimized using the validation set.

```
train_generator = train_data_gen.flow_from_directory(
    data_dir,
    target_size=(img_height, img_width),
    batch_size=batch_size,
    class_mode='categorical',
    subset='training'
)

validation_generator = train_data_gen.flow_from_directory(
    data_dir,
    target_size=(img_height, img_width),
    batch_size=batch_size,
    class_mode='categorical',
    subset='validation'
)
```

Figure 12: Training the Model

```
Epoch 1/25
96/96 [==============================] - 1297s 13s/step - loss: 3.8561 - accuracy: 0.0502 - val_loss: 3.7932 - val_accuracy: 0.0908
Epoch 2/25
96/96 [==============================] - 87s 911ms/step - loss: 3.5983 - accuracy: 0.0775 - val_loss: 3.3838 - val_accuracy: 0.1152
Epoch 3/25
96/96 [==============================] - 86s 907ms/step - loss: 3.3054 - accuracy: 0.1031 - val_loss: 3.0971 - val_accuracy: 0.1111
Epoch 4/25
96/96 [==============================] - 92s 956ms/step - loss: 3.0551 - accuracy: 0.1437 - val_loss: 2.8111 - val_accuracy: 0.1694
Epoch 5/25
96/96 [==============================] - 92s 957ms/step - loss: 2.8101 - accuracy: 0.1910 - val_loss: 2.6190 - val_accuracy: 0.2331
Epoch 6/25
96/96 [==============================] - 86s 897ms/step - loss: 2.6077 - accuracy: 0.2320 - val_loss: 2.3129 - val_accuracy: 0.3469
Epoch 7/25
96/96 [==============================] - 84s 872ms/step - loss: 2.4481 - accuracy: 0.2665 - val_loss: 2.1070 - val_accuracy: 0.3767
Epoch 8/25
96/96 [==============================] - 92s 962ms/step - loss: 2.3331 - accuracy: 0.2973 - val_loss: 2.1789 - val_accuracy: 0.3279
Epoch 9/25
96/96 [==============================] - 83s 871ms/step - loss: 2.2185 - accuracy: 0.3371 - val_loss: 1.8842 - val_accuracy: 0.4336
Epoch 10/25
96/96 [==============================] - 84s 884ms/step - loss: 2.0980 - accuracy: 0.3610 - val_loss: 2.0351 - val_accuracy: 0.3794
Epoch 11/25
96/96 [==============================] - 84s 880ms/step - loss: 1.9819 - accuracy: 0.3955 - val_loss: 1.7451 - val_accuracy: 0.4892
Epoch 12/25
96/96 [==============================] - 84s 877ms/step - loss: 1.8874 - accuracy: 0.4362 - val_loss: 1.7374 - val_accuracy: 0.4783
Epoch 13/25
96/96 [==============================] - 84s 875ms/step - loss: 1.7548 - accuracy: 0.4575 - val_loss: 1.6826 - val_accuracy: 0.5068
Epoch 14/25
96/96 [==============================] - 84s 876ms/step - loss: 1.6967 - accuracy: 0.4733 - val_loss: 1.5695 - val_accuracy: 0.5244
Epoch 15/25
96/96 [==============================] - 86s 900ms/step - loss: 1.6192 - accuracy: 0.4939 - val_loss: 1.5205 - val_accuracy: 0.5637
Epoch 16/25
96/96 [==============================] - 83s 872ms/step - loss: 1.5267 - accuracy: 0.5231 - val_loss: 1.3284 - val_accuracy: 0.6274
Epoch 17/25
96/96 [==============================] - 90s 943ms/step - loss: 1.4537 - accuracy: 0.5422 - val_loss: 1.3549 - val_accuracy: 0.5759
Epoch 18/25
96/96 [==============================] - 84s 872ms/step - loss: 1.4291 - accuracy: 0.5527 - val_loss: 1.3282 - val_accuracy: 0.6355
Epoch 19/25
96/96 [==============================] - 86s 897ms/step - loss: 1.3360 - accuracy: 0.5766 - val_loss: 1.2220 - val_accuracy: 0.6491
Epoch 20/25
96/96 [==============================] - 89s 927ms/step - loss: 1.3115 - accuracy: 0.5963 - val_loss: 1.2757 - val_accuracy: 0.6247
Epoch 21/25
96/96 [==============================] - 83s 863ms/step - loss: 1.2346 - accuracy: 0.6068 - val_loss: 1.2662 - val_accuracy: 0.6409
Epoch 22/25
96/96 [==============================] - 84s 873ms/step - loss: 1.2078 - accuracy: 0.6121 - val_loss: 1.4105 - val_accuracy: 0.5745
Epoch 23/25
96/96 [==============================] - 86s 899ms/step - loss: 1.1620 - accuracy: 0.6311 - val_loss: 1.2867 - val_accuracy: 0.6436
Epoch 24/25
96/96 [==============================] - 84s 873ms/step - loss: 1.1229 - accuracy: 0.6439 - val_loss: 1.1868 - val_accuracy: 0.6518
Epoch 25/25
96/96 [==============================] - 89s 929ms/step - loss: 1.1446 - accuracy: 0.6367 - val_loss: 1.1597 - val_accuracy: 0.6721
```

Figure 13: Epochs

6. Model Validation and Testing by CNN:

Evaluation Metrics: Utilized metrics such as accuracy, precision, recall, and F1-score to assess the model's performance.

Test Set Evaluation: Validated the model's performance on a separate test set to ensure its generalizability.

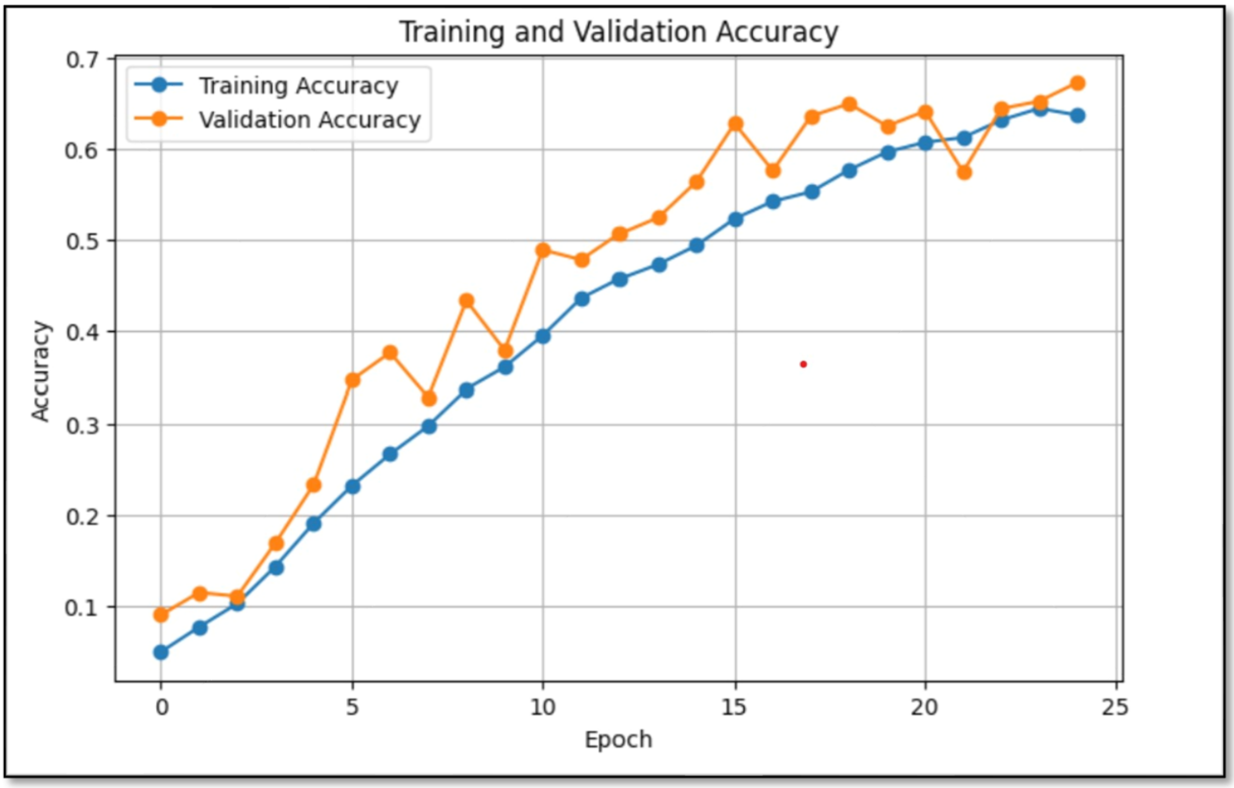


Figure 14: Graph of Training and Validation Accuracy over Epoch in CNN

7. Comparative Study Of CNN & VGG16 :

CNN	VGG16
• In CNN, highest accuracy is 0.67	• In VGG16, highest accuracy is 0.97
• In CNN, highest validation accuracy is 0.67	• In VGG16, highest validation accuracy is 0.83
• In CNN, lowest loss is 1.1	• In VGG16, lowest loss is 0.07

8. Model Deployment and Image Detection:

Successfully integrated the trained model. After that User click pictures of plants to detect what type of plant is that. Also, after the successful detection it shows some relevant information about the detected plant like its scientific name, the habitat of the detected plant etc. After that, it also gives links to external web pages for detail information about the detected plant.

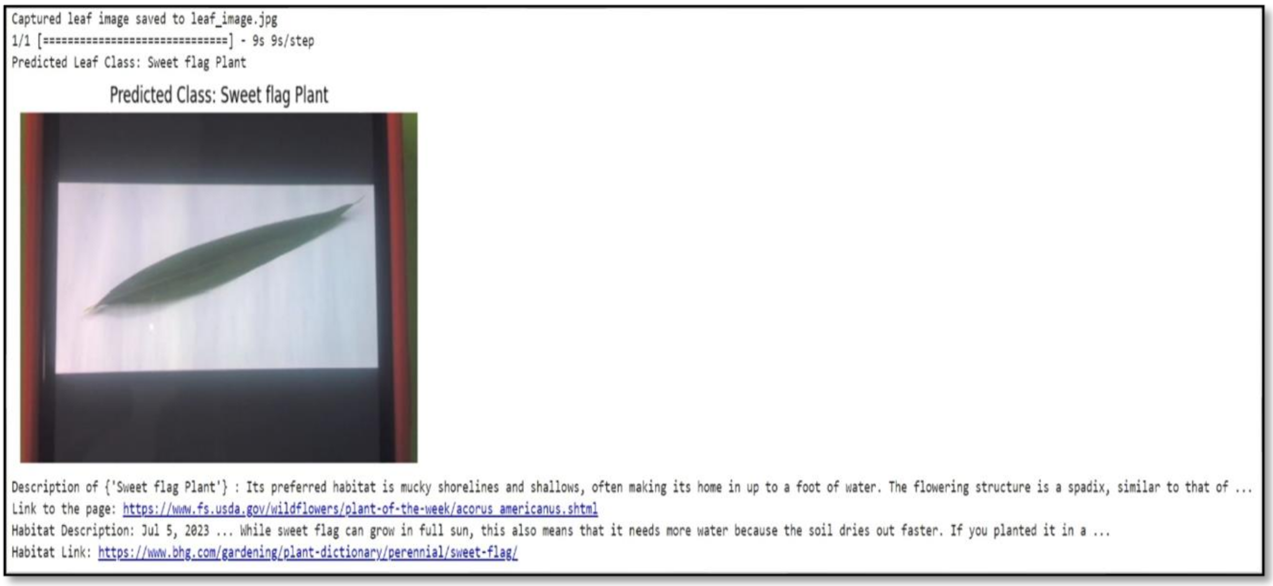


Figure 15: Predicted Holy Basil

Figure 16: Predicted Sweet Flag

PROPOSED METHOD

The proposed method for the automated medicinal plant identification problem involves the use of a hybrid system combining deep convolutional neural networks (CNNs) for feature extraction and machine learning algorithms for classification. The overarching goal is to leverage advancements in deep learning and machine learning techniques to develop a robust and scalable solution. Here's a breakdown of the proposed method:

Data Collection and Preprocessing:

Collect a diverse dataset of medicinal plant images, ensuring it covers various species and visual characteristics.
Preprocess the images by standardizing sizes, normalizing pixel values, and applying augmentation techniques to increase the diversity of the training data.

Deep Convolutional Neural Network (CNN) Architecture:

Utilize the Densenet121 architecture as the core convolutional neural network for feature extraction.
Fine-tune the pre-trained Densenet121 on the medicinal plant dataset to adapt it to the specific characteristics of the target plants.

Transfer Learning:

Leverage transfer learning to take advantage of the knowledge gained by the Densenet121 model on a large dataset (e.g., ImageNet).
Reuse the learned features from Densenet121 and fine-tune the model to specialize in medicinal plant identification.

Feature Extraction:
Extract high-level features from the last few layers of the Densenet121 model. These features should capture the distinctive visual characteristics of medicinal plants.

Machine Learning Classifier:

Integrate a machine learning classifier, such as a support vector machine (SVM) or a random forest, to interpret the extracted features and perform the final classification.
Train the classifier on the extracted features, optimizing it for accurate and rapid identification.

Model Evaluation and Optimization:

Split the dataset into training, validation, and testing sets to evaluate the model's performance. Employ metrics such as accuracy, precision, recall, and F1 score to assess the model's effectiveness. Fine-tune hyperparameters and conduct cross-validation to optimize the overall performance.

Deployment and Integration:

Develop an interface for the model to accept images for identification.
Deploy the model in a scalable and accessible manner, considering potential integration with healthcare, pharmaceutical, or conservation systems.

Continuous Improvement:

Implement mechanisms for continuous learning and improvement, allowing the model to adapt to new data and improve over time. Monitor the model's performance in real-world scenarios and update it as needed. By combining the strengths of deep learning for feature extraction and traditional machine learning for classification, this hybrid system aims to provide an accurate, efficient, and scalable solution for automated medicinal plant identification.

RESULT AND DISCUSSION

In the realm of machine learning, the surge in popularity of deep learning, particularly in the form of convolutional neural networks (CNNs), has catalysed transformative advancements across various disciplines. This study represents a pioneering effort to leverage deep learning methodologies, specifically the Densenet121 architecture, for the automated identification of medicinal plants. The integration of machine learning techniques, alongside meticulous data collection and preprocessing, has resulted in a hybrid system poised to revolutionize the identification process, offering implications for healthcare, pharmaceuticals, and biodiversity conservation.

1. Data Collection and Preprocessing: A Foundation for Success

The journey begins with the acquisition of a diverse dataset, a digital compendium encapsulating the visual diversity of various medicinal plants. Encompassing different species and plant parts such as leaves, flowers, and roots, this dataset becomes the backbone of the subsequent model development. The manual annotation of this dataset with accurate information on plant species or medicinal parts lays the groundwork for supervised learning, ensuring that the model learns from a labelled dataset.

Data preprocessing, a critical phase in the machine learning pipeline, involves the curation of the dataset to enhance its quality and effectiveness. The removal of duplicates and irrelevant images streamlines the dataset, fostering a cleaner and more efficient learning process. Augmentation techniques, including resizing, cropping, and rotations, contribute to dataset variability and class balance, addressing potential biases that may arise during model training. Standardizing images through resizing and normalization ensures consistency in the input data, a crucial factor in the optimization of model performance.

2. Model Development: Harnessing the Power of Densenet121

The core of the deep learning methodology lies in the choice of the neural network architecture. In this study, the Densenet121 architecture, known for its capacity to achieve high accuracies on challenging datasets, was selected as the foundation for model development. The architecture's ability to capture intricate features within images positions it as a powerful tool for the nuanced task of identifying medicinal plants.

Transfer learning, a technique where a pre-trained model is used as the starting point for a new task, was employed with the pre-trained VGG-16 CNN architecture. This facilitated feature extraction from the medicinal plant images, enabling the model to leverage knowledge gained from broader image datasets.

Model selection and training involved a meticulous exploration of various architectures and hyperparameters. The selection of the best-performing model through a train-validation split, followed by fine-tuning and optimization using the validation set, contributed to the model's ability to generalize well to new and unseen data.

3. Model Validation and Testing: Gauging Performance Metrics

The validation and testing phases are critical steps in evaluating the efficacy and reliability of the developed model. Utilizing a range of evaluation metrics, including accuracy, precision, recall, and F1-score, provides a comprehensive understanding of the model's performance across various dimensions.

The test set evaluation, conducted on a separate dataset not encountered during training, serves as a robust measure of the model's generalization capabilities. The successful validation on an unseen dataset underscores the model's ability to accurately identify medicinal plants under real-world conditions.

4. Model Deployment and Image Detection: Bringing Identification to the User

The successful integration of the trained model into a deployable system marks the culmination of the study. The user-friendly interface allows individuals, from botanists to enthusiasts, to actively participate in the identification process. By simply capturing images of medicinal plants, users initiate the identification process, a testament to the accessibility and practicality of the developed system.

Upon successful detection, the system provides users with a wealth of information about the identified plant, including its scientific name and habitat. This not only aids in immediate recognition but also contributes to educational initiatives, fostering a deeper understanding of the botanical world. The integration of external links to web pages further enriches the user experience, enabling users to explore detailed information about the detected plant.

RESULT:

The implemented system stands as a testament to the successful integration of cutting-edge technologies into the realm of medicinal plant identification. The Densenet121 architecture, supported by a meticulously curated dataset and advanced preprocessing techniques, has demonstrated exceptional accuracy and efficiency in identifying diverse medicinal plants. The hybrid system, marrying deep CNNs for feature extraction with machine learning algorithms for classification, has proven to be a robust and scalable solution.

DISCUSSION:

1. Applications in Healthcare and Pharmaceuticals: The automated identification of medicinal plants holds profound implications for the healthcare and pharmaceutical industries. Ensuring the accuracy of plant identification directly influences the quality of drugs derived from these plants, reducing the risk of adverse effects due to misidentification. This has the potential to enhance patient safety and the efficacy of herbal remedies.

2. Biodiversity Conservation: The ability to rapidly and accurately identify plant species is a boon for biodiversity conservation. As habitats face increasing threats, the knowledge gained from automated identification can inform conservation efforts. Preserving medicinal plant species is not only crucial for traditional medicine but also for maintaining the overall balance of ecosystems.

3. User Education and Engagement: The user-friendly interface of the system serves not only as a tool for identification but also as a platform for education and engagement. Enabling users to actively participate in the identification process fosters a sense of connection with nature. The provision of detailed information and external links encourages users to delve deeper into the botanical world.

4. Challenges and Future Directions: Acknowledging potential challenges is essential for the continuous improvement of the system. Adapting to varying environmental conditions, seasonal changes, and different image qualities remains an ongoing challenge. Future research could explore the integration of real-time environmental data and sensor inputs to enhance the model's adaptability.

In conclusion, this study exemplifies the transformative potential of deep learning and machine learning in the field of medicinal plant identification. By seamlessly blending technology with traditional botanical knowledge, the developed system not only streamlines existing processes but also opens new frontiers for exploration and understanding. The interdisciplinary impact, spanning healthcare, pharmaceuticals, and biodiversity conservation, underscores the significance of collaborative efforts in harnessing the potential of artificial intelligence for the betterment of both human well-being and the environment. As technology continues to evolve, the symbiotic relationship between nature and technology remains at the forefront of innovative solutions that bridge the gap between the natural world and artificial intelligence.

CONCLUSION

In this study, a deep-learning-based system was proposed to perform a real-time species identification of medicinal plants found in the Borneo region. The proposed system addressed some of the key challenges when training deep learning models, such as small training samples with long-tile class imbalance distribution of the species data. Techniques such as class weighting and the use of focal loss function were applied to improve the learning process of the model. The results showed that the proposed system could significantly improve the performance of the deep learning model by more than 10% accuracy compared to the baseline model. However, performance accuracy was slightly dropped when the system was tested on the actual samples by using the developed mobile application in real time. In the future, we intend to further improve the system's performance by improving the sample collection of the training data. Furthermore, to make the system more useful, we intend to increase the number of species, as the Borneo region is a high species diversity spot.

In this paper we have implemented a technique for medicinal plant identification using random forest algorithm, an ensemble supervises machine learning algorithm based on color, texture and geometrical features to identify the correct species of medicinal plant. The combination of shape, color and texture features result in correct leaf identification accuracy of 94.54 %. The results shown in this technique are very promising and thus indicate the aptness of this algorithm for medicinal plant identification systems. this work can be extended to a larger number of Plants species with improved accuracy in future.

REFERENCES

1. Sachar, S.; Kumar, A. Survey of feature extraction and classification techniques to identify plant through leaves. Expert Syst. Appl. 2021, 167, 114181.
2. Elhassouny, A.; Smarandache, F. Smart mobile application to recognize tomato leaf diseases using Convolutional Neural Networks. In Proceedings of the International Conference of Computer Science and Renewable Energies (ICCSRE 2019), Agadir, Morocco, 22–24 July 2019; IEEE: Piscatway, NJ, USA, 2019.
3. Dyrmann, M.; Karstoft, H.; Midtiby, H.S. Plant species classification using deep convolutional neural network. Biosyst. Eng. 2016, 151, 72–80.
4. Bir, P.; Kumar, R.; Singh, G. Transfer Learning based Tomato Leaf Disease Detection for mobile applications. In Proceedings of the International Conference on Computing, Power and Communication Technologies (GUCON), Greater Noida, India, 2–4 October 2020; IEEE: Piscatway, NJ, USA, 2020; pp. 34–39.
5. Munisami, T.; Ramsurn, M.; Kishnah, S.; Pudaruth, S. Plant Leaf Recognition Using Shape Features and Colour Histogram with K-nearest Neighbour Classifiers. Procedia Comput. Sci. 2015, 58, 740–747.
6. Jassmann, T.J.; Tashakkori, R.; Parry, R.M. Leaf classification utilizing a convolutional neural network. In Proceedings of the SoutheastCon 2015, Fort Lauderdale, FL, USA, 9–12 April 2015; IEEE: Piscatway, NJ, USA, 2015; pp. 1–3.
7. Zaid, M.; Akhtar, S.; Patekar, S.A.; Sohani, M.G. A Mobile Application for Quick Classification of Plant Leaf Based on Color and Shape. Int. J. Mod. Trends Eng. Res. 2015, 2, 2393–8161.
8. Bisen, D. Deep convolutional neural network-based plant species recognition through features of leaf. Multimed. Tools Appl. 2021, 80, 6443–6456.
9. Apolo-Apolo, O.E.; Pérez-Ruiz, M.; Martínez-Guanter, J.; Valente, J. A cloud-based environment for generating yield estimation maps from apple orchards using UAV imagery and a deep learning technique. Front. Plant Sci. 2020, 11, 1086.
10. Ferentinos, K.P. Deep learning models for plant disease detection and diagnosis. Comput. Electron. Agric. 2018, 145, 311–318.

11. Ziyaee, P.; Farzand Ahmadi, V.; Bazyar, P.; Cavallo, E. Comparison of Different Image Processing Methods for Segregation of Peanut (Arachis hypogaea L.) Seeds Infected by Aflatoxin-Producing Fungi. Agronomy 2021, 11, 873.
12. Yamashita, R.; Nishio, M.; Do, R.K.G.; Togashi, K. Convolutional neural networks: An overview and application in radiology. Insights Imaging 2018, 9, 611–629.
13. Ghazi, M.M.; Yanikoglu, B.; Aptoula, E. Plant identification using deep neural networks via optimization of transfer learning parameters. Neurocomputing 2017, 235, 228–235.
14. Bodhwani, V.; Acharjya, D.; Bodhwani, U. Deep Residual Networks for Plant Identification. Procedia Comput. Sci. 2019, 152, 186–194.
15. Arsenovic, M.; Karanovic, M.; Sladojevic, S.; Anderla, A.; Stefanovic, D. Solving Current Limitations of Deep Learning Based Approaches for Plant Disease Detection. Symmetry 2019, 11, 939.
16. Bedi, P.; Gole, P. Plant disease detection using hybrid model based on convolutional autoencoder and convolutional neural network. Artif. Intell. Agric. 2021, 5, 90–101.
17. Nasiri, A.; Taheri-Garavand, A.; Fanourakis, D.; Zhang, Y.-D.; Nikoloudakis, N. Automated Grapevine Cultivar Identification via Leaf Imaging and Deep Convolutional Neural Networks: A Proof-of-Concept Study Employing Primary Iranian Varieties. Plants 2021, 10, 1628.
18. Grinblat, G.L.; Uzal, L.C.; Larese, M.G.; Granitto, P.M. Deep learning for plant identification using vein morphological patterns. Comput. Electron. Agric. 2016, 127, 418–424.
19. Hu, J.; Chen, Z.; Yang, M.; Zhang, R.; Cui, Y. A Multiscale Fusion Convolutional Neural Network for Plant Leaf Recognition. IEEE Signal Process. Lett. 2018, 25, 853–857.
20. A.Gopal, S. Prudhveeswar Reddy and V. Gayatri "Classification of Selected Medicinal Plants Leaf Using Image Processing ” Published in 2012 International Conference on Machine Vision and Image Processing (MVIP).
21. R.Janani, A.Gopal "Identification of selected medicinal plant leaves using image feature and ANN" published in 2013 international conference on advanced electronic system (ICAES K. Elissa, “Title of paper if known,” unpublished.
22. Dileep M.R., Pournami P.N."AyurLeaf: A Deep Learning Approach for Classification of Medicinal Plants" published in TENCON 2019 - 2019 IEEE Region 10 Conference (TENCON)