A Project Report

on

MEDICINAL PLANTS CLASSIFICATION USING CONVOLUTION NEURAL NETWORK(CNN)

Submitted in Partial Fulfillment of the Requirements for the Award of Degree of

BACHELOR OF TECHNOLOGY

In

ARTIFICIAL INTELLIGENCE & MACHINE LEARNING

By

N. Deepak Sai	21K61A6142
K. Pravallika	21K61A6129
K. Varshini	21K61A6126
M. Ajay	21K61A6136

Under the esteemed guidance of

Dr. SHAIK MOHAMMAD RAFEE

HOD & Professor, AI & ML

...Values We Live...
sasi autonomous
INSTITUTE OF TECHNOLOGY & ENGINEERING
TADEPALLIGUDEM
Accredited by NAAC with "A+" Grade
Recognised by UGC under section 2(f) &12(B)
Approved by AICTE - New Delhi
Permanently Affiliated to JNTUK, SBTET
Ranked as "A" Grade by Govt. of A.P.
Tel : 08818 275500, Email : office@sasi.ac.in, Website :www.sasi.ac.in
SITE : Near Aerodrome, TADEPALLIGUDEM - 534 101, West Godavari District, Andhra Pradesh.

DEPARTMENT OF ARTIFICIAL INTELLIGENCE & MACHINE LEARNING

ACADEMIC YEAR 2024-2025

VISION AND MISSION OF INSTITUTE

VISION

Confect as a premier institute for professional education by creating technocrats who can address the society's needs through inventions and innovations.

MISSION

Partake in the national growth of technological, industrial, and industrial areas with societal responsibilities.

Provide an environment that promotes productive research.

Meet stakeholder's expectations through continued and sustained quality improvements.

VISION AND MISSION OF DEPARTMENT

VISION

To create competent Engineers with solid foundation in the domain of Artificial Intelligence and Machine Learning to contribute in uplifting of rural communities.

MISSION

To create an ambience that facilitates blended learning and extensive use of emerging technologies as a path forward to meaningful employment in the field of artificial intelligence and machine learning.

To facilitate collaborative learning with multi-disciplinary teams that encourage research initiatives leading to innovations.

To inculcate among students the highest level of professional conduct and ethical values.

PROGRAM OUTCOMES (POs)

PO1: Engineering Knowledge

Apply knowledge of mathematics, natural science, computing, engineering fundamentals and an engineering specialization as specified in WK1 to WK4 respectively to develop to the solution of complex engineering problems.

PO2: Problem Analysis

Identify, formulate, review research literature and analyze complex engineering problems reaching substantiated conclusions with consideration for sustainable development.

PO3: Design/Development of Solutions

Design creative solutions for complex engineering problems and design/develop systems, components, or processes to meet identified needs with consideration for the public health and safety, whole-life cost, net zero carbon, culture, society, and environment as required.

PO4: Conduct Investigations of Complex Problems

Conduct investigations of complex engineering problems using research-based knowledge including design of experiments, modelling, analysis, and interpretation of data to provide valid conclusions.

PO5: Engineering Tool Usage

Create, select and apply appropriate techniques, resources, and modern engineering and IT tools, including prediction and modelling, recognizing their limitations to solve complex engineering problems.

PO6: The Engineer and the World

Analyze and evaluate societal and environmental aspects while solving complex engineering problems for their impact on sustainability with reference to economy, health, safety, legal framework, culture, and environment.

PO7: Ethics

Apply ethical principles and commit to professional ethics, human values, diversity and inclusion; adhere to national and international laws.

PO8: Individual and Collaborative Team Work

Function effectively as an individual, and as a member or leader in diverse/multidisciplinary teams.

PO9: Communication

Communicate effectively and inclusively within the engineering community and society at large, such as being able to comprehend and write effective reports and design documentation, and make effective presentations considering cultural, language, and learning differences.

PO10: Project Management and Finance

Apply knowledge and understanding of engineering management principles and economic decision-making and apply these to one's own work, as a member and leader in a team, and to manage projects in multidisciplinary environments.

PO11: Life-Long Learning

Recognize the need for, and have the preparation and ability for independent and life-long learning, adaptability to new and emerging technologies, and critical thinking in the broadest context of technological change.

PROGRAM EDUCATIONAL OBJECTIVES (PEOS)

PEO1:

Graduates will be able to apply the domain knowledge and the technological skills to gain meaningful employment and adapt to the ever demand of technological landscape.

PEO2:

Graduates will be able to pursue and excel in higher education and research.

PEO3:

Graduates will be able to evolve as leaders exhibiting highest level of ethics.

PROGRAM SPECIFIC OUTCOMES (PSOS)

PSO1:

Students will be able to utilize core principles of Artificial Intelligence Engineering for the design, development and prototyping of AI Subsystems.

PSO2:

Students will be able to employ acquired knowledge in data storage, data analytics, and Machine Intelligence to address and solve practical business challenges.

EXPECTED OUTCOMES

PROGRAM OUTCOMES (POs)

PO1: Engineering Knowledge

PO2: Problem Analysis

PO3: Design/Development of Solutions

PO4: Conduct Investigations of Complex Problems

PO5: Engineering Tool Usage

PO6: The Engineer and The World PO7: Ethics

PO8: Individual and Collaborative Team Work PO9: Communication

PO10: Project Management and Finance PO11: Life-Long Learning

PROGRAM SPECIFIC OUTCOME (PSOs)

PSO1: Applicable

PSO2: Applicable

SASI INSTITUTE OF TECHNOLOGY & ENGINEERING
autonomous TADEPALLIGUDEM

Accredited by NAAC with "A+" Grade
Recognised by UGC under section 2(f) &12(B)
Approved by AICTE - New Delhi
Permanently Affiliated to JNTUK, SBTET
Ranked as "A" Grade by Govt. of A.P.

Tel : 08818 275500, Email : office@sasi.ac.in, Website :www.sasi.ac.in
SITE : Near Aerodrome, TADEPALLIGUDEM - 534 101, West Godavari District, Andhra Pradesh.

DEPARTMENT OF ARTIFICIAL INTELLIGENCE & MACHINE LEARNING

CERTIFICATE

This is to certify that the project work entitled "MEDICINAL PLANTS CLASSIFICATION USING CONVOLUTION NEURAL NETWORK(CNN)" is being submitted by N. Deepak Sai (21K61A6142), K. Pravallika (21K61A6129), K. Varshini (21K61A6126), M. Ajay (21K61A6136) in partial fulfillment for the award of the degree of **BACHELOR OF TECHNOLOGY, in Artificial Intelligence & Machine Learning** to Jawaharlal Nehru Technological University, Kakinada during the academic year 2024 to 2025 is a record of Bonafide work carried out by them under my/our guidance and supervision. The results presented in this thesis have been verified and are found to be satisfactory. The results embodied in this thesis have not been submitted to any other University or Institute for the award of any other degree or diploma.

Project Supervisor
Dr. Shaik Mohammad Rafee
Professor & HOD
Department of AI & ML

Head of the Department
Dr. Shaik Mohammad Rafee
Professor & HOD
Department of AI & ML

External Examiner

sasi INSTITUTE OF TECHNOLOGY & ENGINEERING
autonomous TADEPALLIGUDEM
...Values We Live...
Accredited by NAAC with "A+" Grade
Recognised by UGC under section 2(f) &12(B)
Approved by AICTE - New Delhi
Permanently Affiliated to JNTUK, SBTET
Ranked as "A" Grade by Govt. of A.P.
Tel : 08818 275500, Email : office@sasi.ac.in, Website :www.sasi.ac.in
SITE : Near Aerodrome, TADEPALLIGUDEM - 534 101, West Godavari District, Andhra Pradesh.

DEPARTMENT OF ARTIFICIAL INTELLIGENCE & MACHINE LEARNING

DECLARATION BY THE CANDIDATES

We **N. Deepak Sai (21K61A6142), K. Pravallika (21K61A6129), K. Varshini (21K61A6126), M. Ajay (21K61A6136)**, here by declare that the project report entitled "**MEDICINAL PLANTS CLASSIFICATION USING CONVOLUTION NEURAL NETWORK(CNN)**" carried out under esteemed supervision of **Dr. Shaik Mohammad Rafee**, is submitted in partial fulfillment of the requirements for the award of the degree of Bachelor of Technology in Artificial Intelligence & Machine Learning. This is a record of work carried out by us and the results embodied in this project has not been reproduced or copied from any source. The results embodied in this project report have not been submitted to any other University or Institute for the award of any other degree or diploma.

Project Associates

N. Deepak Sai	21K61A6142
K. Pravallika	21K61A6129
K. Varshini	21K61A6126
M. Ajay	21K61A6136

ACKNOWLEDGEMENT

We pay obeisance to our dynamic Chairman, **Sri B. Venu Gopala Krishna**, Sasi Educational Society, for his inspiring presence, which has always been the principal driving force behind all-over endeavors. First of all, we would like to extend a special thanks to Sri. M. Narendra Krishna, Vice- Chairman, Sasi Institute of Technology and Engineering, for his everlasting support.

It is true pleasure for us to thank **Dr. Mohammad Ismail**, Principal, Sasi Institute of Technology and Engineering, who is the striving force for us to make this project through periodical audits.

We feel the privilege to thank **Dr. Shaik Mohammad Rafee**, Professor and Head of the Department, of Artificial Intelligence & Machine Learning, for providing us with invaluable feedback on our work, which allowed us to constantly improve our project.

We are very much grateful to thank our supervisor **Dr. Shaik Mohammad Rafee**, Professor and Head of the Department, of Artificial Intelligence & Machine Learning for his constant encouragement, monitoring, and guidance and live with us throughout the submission of the project. He motivated us whenever we encountered an obstacle along the way.

We are very happy to thank our project coordinator **Mrs. P. Sheela**, and all the Project Evaluation Committee members who provide valuable suggestions to fine- tune our ideas and projects. We are also thankful for all teaching and non-teaching staff members who contributed well to the successful completion of our project work.

With gratitude,

N. Deepak Sai 21K61A6142

K. Pravallika 21K61A6129

K. Varshini 21K61A6126

M. Ajay 21K61A6136

ABSTRACT

Traditional medicine has used medicinal plants for ages as all-natural treatments for a wide range of illnesses. They include medicinal bioactive substances that can be utilized to treat a variety of ailments. The identification and characterization of medicinal plants are of increasing importance due to the rising demand for natural products and the requirement for sustainable healthcare. Based on their physical and chemical features, machine learning (ML) and deep learning (DL) algorithms have shown considerable promise for the detection and classification of therapeutic plants. Natural chemicals found in medicinal plants are a great source for the creation of novel medications and treatments. However, because there are so many diverse species with comparable physical characteristics, it can be difficult to identify and characterize therapeutic plants. Additionally, the habitat, climate, and growing circumstances all have an impact on the chemical makeup of medicinal plants. Therefore, for medicinal plants to be used effectively in medicine, correct identification and classification are essential. When it comes to the identification and classification of medicinal plants, ML and DL approaches have shown considerable potential. These techniques can analyze big datasets and extract features that are difficult for the human eye to see. For instance, image recognition algorithms may examine photos of medicinal plants and pinpoint their distinctive morphological characteristics, such as the size, shape, and texture of their leaves. These characteristics can then be used by classification algorithms to divide medicinal plants into several groups according to their morphological or chemical characteristics. Convolutional neural networks (CNNs) for picture identification are one example of the employment of ML and DL techniques in medicinal plant detection. CNNs are DL models that can spot patterns and distinguish important details in images. CNNs have been used by researchers VIII to categorize many medicinal plants according to the shape, texture, and color of their leaves. The detection and classification of medicinal plants based on their morphological and chemical properties has shown tremendous promise for ML and DL approaches, in conclusion. These techniques can analyze big datasets and extract features that are difficult for the human eye to see feature extraction and image recognition.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVATIONS

CNN	CONVOLUTIONAL NEURAL NETWORK.
FCH	FUZZY LOCAL HISTOGRAM.
FLBP	FUZZY LOCAL BINARY PATTERN.
GLCM	GRAY LEVEL CO-OCCURENCE MATRIX.
PNN	PROBABLISTIC NEURAL NETWORK.
CMF	C- MEANS FUZZY.
SVM	SUPPORT VECTOR MACHINE.

CHAPTER 1

INTRODUCTION

1.1 overview of the project

Plant ID has a significant job in current logical issues, for example, biodiversity, environment, and pharmacology among others. In Biology, plant distinguishing proof includes breaking down numerous organs, for example, blossoms, seeds, leaves and woody parts. This methodology renders the errand troublesome as blossoms and seeds, which are occasional and subject to the plant's age and condition, are difficult to discover. In exceptional circumstances, for example, discovering fossils or uncommon plants, the material accessible to distinguish a plant is only the leaves. To comprehend these circumstances, a leaf morphological scientific categorization method is proposed, which considers just the leaves to play out the distinguishing proof errands [14]. This methodology joins different highlights of leaves, for example, shape, vein structure, surface, and some histological data. Recognizing plants is a troublesome and complex undertaking because of the idea of the leaves. Although that the leaves present some major highlights, they additionally present a wide example of the variety. This variety may happen in various leaves from a similar plant, where attributes, for example, development and presentation to the sun produce varieties in the size, shading, surface, and state of the leaves. These varieties are additionally present in leaves from similar species, however from various plants. Right now, are an outcome of soil impact, atmosphere or even condition when the leaf is being shaped.

Plants have been utilized as medicines for thousands of years in different countries and are a source of many potent and powerful drugs worldwide, a total of more than 35,000 plant species are used for medicinal purposes. The value of medicinal plants to human livelihoods is essentially infinite. The World Health Organization estimated that 80% of the population of de-veloping countries relies on traditional medicines, mostly plant drugs, for their primary healthcare needs. Since time immemorial man has used various parts of plants in the treatment and prevention of many ailments. From prehistoric days, plants are used for shelter, food and medicine. The use of plants for medicinal purposes is as old as our civilization. The first known written record of curative plants was of Sumerian herbal of 2200 BC. In the 5th century BC, The Greek doctor Hippocrates list out some 400 herbs in common use (Lakshmi V et al., 2006). Dioscorides, in the 1st century AD, wrote an herbal by using 600 plants which ultimately became the base for many later works. The World Health Organization estimated that 80% of the population of developing countries religon traditional medicines, mostly plant drugs, for their primary health care needs. Time immemorial man has used various parts of plants in the treatment and prevention of many ailments

The frames have grown so far using programmed order procedures, however the procedures are very comparative. These means include the configuration of the collected leaves, the realization of some previous managements to distinguish their particular characteristics, the arrangement of the leaves, the compilation of the database, the preparation for the recognition and evaluation of the results. The world has a more number of plant species, a significant amount of which have remedial qualities, close to elimination and others are destructive to humans. The recognition of dark plants depends very much on the intrinsic information of a specialized botanist. The best technique for distinguishing plants effectively and effectively is a manual processing strategy based on morphological qualities. In this way, a large number of procedures relating to the organization of these plant species "depend on the accumulation of information and the skills of individuals" [1].Be that as it may, this procedure of manual acknowledgment is regularly relentless and time consuming. Henceforth numerous scientists have directed examinations to help the programmed grouping of plants dependent on their physical qualities.

The frames have grown so far using programmed order procedures,

however the procedures are very comparative. These means include the configuration of the collected leaves, the realization of some previous managements to distinguish their particular characteristics, the arrangement of the leaves, the compilation of the database, the preparation for the recognition and evaluation of the results. A computerized plant distinguishing proof framework can be utilized by non-botanical specialists to rapidly recognize plant species easily.

Normally customary individuals are doled out with the activity of gathering the plants from the timberlands. Once in a while they couldn't perceive the uncommon and significant plants due to human blunder. These uncommon sorts of plants are critical to spare the life of a patient. Additionally, here and there these individuals could get off base species which might be hurtful plants. In such cases, it is important to utilize the programmed plant acknowledgment framework. This framework helps a conventional people or any layman to perceive the diverse plant species. These sorts of frameworks are likewise extremely accommodating for the trekking individuals if they are intrigued to gather the plant species while trekking the mountains.

1.2 Existing method

This research proposed a new mobile application based on Android operating system for identifying Indonesian medicinal plant images based on texture and color features of digital leaf images. In the experiments we used 51 species of Indonesian medicinal plants and each species consists of 48 images, so the total images used in this research are 2,448 images. This research investigates effectiveness of the fusion between the Fuzzy Local Binary Pattern (FLBP) and the Fuzzy Color Histogram (FCH) in order to identify medicinal plants. The FLBP method is used for extracting leaf image texture. The FCH method is used for extracting leaf image color. The fusion of FLBP and FCH is done by using Product Decision Rules (PDR) method. This research used Probabilistic Neural Network (PNN) classifier for classifying medicinal plant species. The experimental results show that the fusion between FLBP and FCH can improve the average accuracy of medicinal plants identification. The accuracy of identification using fusion of FLBP and FCH is 74.51%. This application is very important to help people identifying and finding information about Indonesian medicinal plant. Historically all medicinal preparations were derived from plants, whether in the simple form of plant parts or in the more complex form of crude extracts, mixtures, etc. Today a substantial number of drugs are developed from plants (Fabricant and Farnsworth, 2001) which are active against a number of diseases. The majority of these involve the isolation of the active ingredients found in a particular medicinal plant and its subsequent modification. In the developed countries, 25 percent of the medical drugs are based on plants and their derivatives and the use of medicinal plants is well known among the indigenous people in rural areas of many developing countries.

Disadvantages

The most of existing methods has ignored the poor quality images like images with noise or poor brightness leads to less accuracy.

1.3 Proposed method

The proposed technique was tested on a dataset of 60 medicinal plants classes and each plant class has 20 images. A very high accuracy of 98.3% was obtained with a CNN classifier. The size of each image was 256*256 pixels. Proposed an approach based on fractal dimension features based on leaf shape and vein patterns for the recognition and classification plant leaves. Using a volumetric fractal dimension approach to generate a texture signature for a leaf and the GLCM (Gray level co occurance matrix) algorithm.

Advantages

High accuracy is obtained and time consumption for detecting the shape.

More datasets are included.

CHAPTER 2

LITERATURE SURVEY

Some tests have been performed to create tools for distinctive tests on plants in the past 10 years. Wu et al. They completed one of the most definitive works in the field of plant disposal. [2] From five fundamental geometric aspects, twelve morphological aspects are deduced and, therefore, the analysis of the main components (PCA) is used to reduce the measurement in order to send fewer sources of information to a probabilistic neural system (PNN). They achieved normal accuracy of 90.3% with the Flavia dataset, which is its creation. Using an alternative dataset but a similar classifier, Hossain et al [4] have achieved a degree of accuracy comparable to the comparative highlights [4].

However, using comparable evidence, an alternative dataset with only 20 species, Du et al [5] reached 93% with the closest k classifier [5]. Using another separation measure called "ISO map", we obtain 92.3% accuracy in a dataset of 2000 images containing 20 unique types of sheets [6]. Herdiyeniet et al. They used -9a combination of an example of an almost spongy couplet and a spongy shading histogram and a probabilistic neural system classifier (PNN) in a data set of 2448 leaf images (270 * 240 pixels) acquired from therapeutic plants in the forest Indonesia will achieve a grouping accuracy of 74.5% [7]. Prasvita et al [7] have created a versatile comparative application that depends on previous research [8]. Using the parts descriptor (KDES) as another component extraction strategy, Le et al [9] built a recognizable test framework for a fully mechanized implant. The proposed procedure was tested on a dataset of 55 therapeutic plants in Vietnam and a high accuracy of 98.3% was achieved with the help vector machine classifier (SVM). In addition, its calculation achieved 98.5% accuracy in the Flavia dataset, which is the best result distributed so far in this dataset. Using discrete wavelet modification to eliminate the interpretation of invariant reflections from a variety of 8 distinctive elaborate plants in Indonesia, Arai et al [10] achieved 95.8% accuracy using a classifier of help vector machines (SVM) [10].

The size of each image was 256 * 256 pixels. Du et al [11] proposed a methodology that depends on the measurement of the fractal highlighted on the basis of the shape of the leaf and the designs of the veins for the recognition and order of the leaves of the plant [11]. Using a closer k classifier with 20 highlights, they had the opportunity to achieve a high recognition rate of 87.1%. Using a form of volumetric fractal measurement to try to create a surface mark for a sheet and calculating the linear discriminant analysis (LDA), Backes et al [12] had the opportunity to overcome the conventional methodologies that depended on the Gabor channels and the Fourier test achieved 87.3% accuracy in a data set of 640 leaves from 32 distinctive plant species [13]. They used to model and combine data as it were. The images were obtained using a camera for cell phones with a 1980 * 1024 lens. Hernández-Serna et al [14] achieved an accuracy level of 92.9% using the Flavia dataset [14]. Sixteen sources of information (6 geometric aspects, 8 surfaces and 2 morphological highlights) were sent to a counterfeit neuronal system (ANN) with 60 centers in the covered layer and a learning frequency of over 0.100 pages. Using the equivalent data set, Chaki et al [15] achieved an overall accuracy of 97.6% using a Neuro-Fuzzy (NFC) classifier with a 44- component surface vector and a component shape vector [15]. Use of the module only includes the Flavia and Pattern Net data set (a type of neural system). His neuronal advancement system had two hidden layers with 26 neurons each and was prepared for more than 100 pages.

Fascinating work has been done. They performed the calculation of the scale curvature histogram (HCoS) to extract the shape data and the modification of the almost corresponding example (LBPV) to eliminate the surface data. In the best case, the clean dataset exceeded the 7.3% dataset. This recommends that legitimately acquired images using a mobile phone can create good levels of proven accuracy and images that are physically prepared in a laboratory and then grouped together. Previously, Amin et al [18] used an appropriate neuronal level diagram (DHGN) to acquire fold data using 64 element vectors and the closest k classifier with Canberra separation to acquire 71.5% accuracy [18]. A study of the 25 best-selling pharmaceutical drugs in 1997 found that 11 of them (42%)

were either biological, natural products or entities derived from natural products, with a total value of 17.5 billion US $. The total sales value of drugs (such as Taxol) derived from just one plant species (Taxus baccata) was 2.3 billion US$ in 2000. The world market for herbal remedies in 1999 was calculated to be worth

19.4 billion US$, with Europe in the lead (US$ 6.7 bil-lion), followed by Asia (US$ 5.1 billion), North America (US$ 4.0 billion), Japan (US$ 2.2 bil-lion), and then the rest of the world (US$ 1.4 billion) (Hamilton, 2004).

According to a survey by NCI, USA, 61% of the 877 small-molecules as new chemical entities introduced as drugs worldwide during 1981–2002 were inspired by natural products (Newman et al., 2000). Plant species still serve as a rich source of many novel biologically active compounds; very few plant species have been thoroughly investigated for their medicinal properties (Heinrich and Gibbons, 2001). Thus, there is a renewed interest in phytomedicine during last decade and now a day many medicinal plant species are being screened for pharmacological activities. The use of plants as therapeutic tools, especially those used to relieve Sub-Acute pathologies, have had a remarkable role in the popular medicine of different countries. Many indigenous drugs have been used by practitioners for the treatment of DM throughout the world (Bailey & Day, 1989). However, only a few have received scientific or medical scrutiny and the World Health Organization has recommended accordingly that those traditional plant treatments for diabetes warrant further evaluation (WHO, 1980). Several hundred plants are known to have antidiabetic properties and a large number of compounds from plant extracts have been reported to have beneficial effects for treatment of diabetes. But, the use of medicinal plants in modern medicine suffers from the fact that though hundreds of plants are used in the world to prevent orto cure diseases, scientific evidence in terms of modern medicine is lacking in most cases. Thus, identification of potential antidiabetic agents using mechanism-based studies holds great promisefor elucidating mechanisms and devising more specific and effective treatments for diabetes related diseases (Izzo & Ernst, 2001).The plant families, including the species (sp), most studied for their confirmed hypoglycemic effects include: Leguminoseae (11 sp), Lamiaceae (7 sp), Liliaceae (8 sp), Cucurbitaceae(7 sp), Asteraceae (6 sp), Moraceae (6 sp), Rosaceae (6 sp), Euphorbiaceae (5 sp) and Araliaceae(5 sp). The most studied species are: Citrullus colocynthis (Opuntia streptacantha Lem. (Cactaceae), Trigonella foenum greacum L.(Leguminosea), Momordica charantia L. (Cucurbitaceae),Ficus bengalensis L. (Moraceae), Polygala senega L. (Polygalaceae), and Gymnema sylvestre R.(Asclepiadaceae). Many studies have confirmed the benefits of medicinal plants with hypogly-cemic effects in the management of diabetes mellitus. The effects of these plants may delay the development of diabetic complications and correct the metabolic abnormalities. The native people of Canada use 16 plants for the treatment of diabetes from very ancient period. In

Mexico, there are 306 plants recorded as anti-diabetic plants and the local traditional herbal healers in south eastern Morocco (Tafilalet region), use 92 plants for the treatment of diabetes mellitus and their complications. In the Chinese traditional medical treatment of diabetes, there are hundreds of prescriptions to aim directly at different symptoms of diabetes, most of which come from plants.

Parag Bhandarkar, Rizwan Ahmed et al decomposed the morphology of leaf edges using predefined structural elements and extracted a structural signature which quantifies the leaf shape feature. They used the root mean square error between the feature vectors of the input image and the image in the database for computing the identity. The database created by the authors consists of 40 leaf samples of 10 different species. They achieved an overall classification rate of 67.5%, which is independent of leaf size and orientation. The identification rate is comparatively low to be of use in practical implementations.

Reference	Features	Classifier	Accuracy (%)	Dataset	Training	Testing	Species
Arai *et al.* (2013)	Wavelets	SVM	95.8	120	96	24	8
Hernandez-Serna and Jimenez-Segura (2014)	Shape, Texture	ANN	92.9	1800	1620	180	32
Le *et al.* (2014)	Kernel Descriptor	SVM	98.5	1905	1585	320	32
			98.3	1312	649	663	55
Munisami *et al.* (2015)	Shape, Colour	kNN	87.3	640	Leave-one-out cross-validation		32
Chaki *et al.* (2015)	Shape, Texture	NFC	97.6	930	310	620	31
Siravenha and Carvalho (2015)	Shape	ANN	97.5	1865	Ten-fold cross-validation		32
Carranza-Rojas and Mata-Montero (2016)	Curvature, Texture	kNN	87.2	2345	Leave-one-out cross-validation		66

Fig 1:Existing Models Accuracy Table

2.1 Problem Statement

The identification and classification of medicinal plants have been a challenging task for experts due to the large number of plant species, variations in plant morphology, and environmental factors. Traditional methods of identification, such as field observations and taxonomic keys, are time-consuming, expensive, and require extensive botanical knowledge. With the increasing demand for medicinal plants due to their potential health benefits and the need for sustainable and responsible harvesting practices, the development of an accurate and efficient system using Machine Learning (ML) and Deep Learning (DL) algorithms to identify and classify different types of medicinal plants based on their images is essential. This system will be developed using advanced ML and DL algorithms that can analyze visual features and patterns in plant images to identify and classify different types of medicinal plants. The system will be trained using a large dataset of images of different medicinal plant species, along with their corresponding labels. The images will be preprocessed to remove any noise and enhance the features that are important for classification. The ML and DL algorithms used in this system will be capable of recognizing complex patterns in the images, enabling accurate identification and classification of medicinal plants. The system will also be able to adapt to variations in plant morphology and environmental factors, making it more robust and accurate. The system will be developed using open-source ML and DL frameworks such as TensorFlow, PyTorch, and Keras, making it easy to customize and integrate into existing applications. The development of this system will have a significant impact on various fields, such as traditional medicine, agriculture, and biodiversity conservation. Experts in traditional medicine will be able to easily and quickly identify medicinal plants, which is essential for their proper use in traditional medicine and drug discovery. The system will also be useful in the agricultural industry, where it can help farmers identify and classify different types of medicinal plants, enabling them to make better decisions about their cultivation and 6 harvesting practices. Additionally, the system can aid in biodiversity conservation efforts by enabling experts to accurately identify and classify different types of medicinal plants in natural habitats. In conclusion, the development of an accurate and efficient system using ML and DL algorithms to identify and classify different types of medicinal plants based on their images is a crucial step towards sustainable and responsible harvesting practices. This system will enable experts in traditional medicine, agriculture, and biodiversity conservation to easily and quickly identify medicinal plants, which is essential for their proper use in traditional medicine and drug discovery.

2.2 Objectives

Developing an efficient and accurate method for identifying medicinal plant species based on their leaf images is a crucial step towards sustainable harvesting practices. To achieve this goal, the collected images need to be preprocessed and data augmentation techniques should be employed to increase the size and diversity of the dataset. The dataset should contain images of various medicinal plant species, each with their corresponding label. To develop a deep learning model for detecting medicinal plants, advanced algorithms like convolutional neural networks (CNNs) can be utilized. These models can learn complex features and patterns in the images, making them ideal for plant identification tasks. The goal is to develop an automated and efficient method for accurately identifying and classifying different plant species based on unique leaf characteristics. Once the model is developed, it needs to be evaluated using a validation set and various metrics such as accuracy, precision, recall, and F1 score. The performance of the model can be further improved by fine-tuning the hyperparameters and using transfer learning techniques. For instance a pre-trained CNN model, can be used as a base model, and its weights can be fine-tuned to improve the accuracy of the medicinal plant classification task. After training and validating the model, it can be used to classify new images of medicinal plant leaves. A user-friendly web application can be developed for this purpose, where users can easily upload images of plant leaves and get instant results. The web application can be developed using popular web frameworks like Flask or Django, along with front-end libraries like React. In conclusion, the development of an automated and accurate method for identifying medicinal plant species based on their leaf images is an essential step towards sustainable harvesting practices. By utilizing deep learning models, pre-processing techniques, and data augmentation, we can develop a model that is robust and efficient in identifying various medicinal plant species. The use of transfer learning techniques can also improve the performance of the model. With the development of a user-friendly web application, experts in traditional medicine, agriculture, and biodiversity conservation can easily and quickly identify medicinal plants.

CHAPTER 3
METHODOLOGY

3.1 Aim of the project

The main aim of this project is to automatically recognize the medicinal plants. Using the base features which are extracted directly from the image, a number of derived features are calculated. Ratios are more suitable for comparison as they are independent of the actual size of the image in pixels.

3.2 System Requirements

3.2.1 Software Requirements

Operating System: WINDOWS 7,8,10

Software: PYTHON 3.13.3

Data Base: SQL

3.2.2 Hardware Requirements

CPU type: Intel Pentium 4

Clock speed: 3.0 GHz

Ram size: 512 MB

CD -drive type : 52xmax

3.3 Overview Of The Platform

3.3.1 Python

Python is a widely used general-purpose, high level programming language. It was initially designed by Guido van Rossum in 1991 and developed by Python Software Foundation. It was mainly developed for emphasis on code readability, and its syntax allows programmers to express concepts in fewer lines of code.

Python is a programming language that lets you work quickly and integrate systems more efficiently.

It is used for:

web development (server-side),

software development,

mathematics,

System scripting.

What can Python do?

Python can be used on a server to create web applications.

Python can be used alongside software to create workflows.

Python can connect to database systems. It can also read and modify files.

Python can be used to handle big data and perform complex mathematics.

Python can be used for rapid prototyping, or for production-ready software development.

Why Python?

Python works on different platforms (Windows, Mac, Linux, Raspberry Pi, etc).

Python has a simple syntax similar to the English language.

Python has syntax that allows developers to write programs with fewer lines than some other programming languages.

Python runs on an interpreter system, meaning that code can be executed as soon as it is written. This means that prototyping can be very quick.

Python can be treated in a procedural way, an object-orientated way or a functional way.

Good to know

The most recent major version of Python is Python 3, which we shall be using in this tutorial. However, Python 2, although not being updated with anything other than security updates, is still quite popular.

Python 2.0 was released in 2000, and the 2.x versions were the prevalent releases until December 2008. At that time, the development team made the decision to release version 3.0, which contained a few relatively small but significant changes that were not backward compatible with the 2.x versions. Python 2 and 3 are very similar, and some features of Python 3 have been backported to Python 2. But in general, they remain not quite compatible.

Both Python 2 and 3 have continued to be maintained and developed, with periodic release updates for both. As of this writing, the most recent versions available are 2.7.15 and 3.6.5. However, an official End of Life date of January 1, 2020 has been established for Python 2, after which time it will no longer be maintained.

Python is still maintained by a core development team at the Institute, and Guido is still in charge, having been given the title of BDFL (Benevolent Dictator For Life) by the Python community. The name Python, by the way, derives not from the snake, but from the British comedy troupe Monty Python's Flying Circus, of which Guido was, and presumably still is, a fan. It is common to find references to Monty Python sketches and movies scattered throughout the Python documentation.

It is possible to write Python in an Integrated Development Environment, such as Thonny, Pycharm, Netbeans or Eclipse which are particularly useful when managing larger collections of Python files.

Python Syntax compared to other programming languages

Python was designed to for readability, and has some similarities to the English language with influence from mathematics.

Python uses new lines to complete a command, as opposed to other programming languages which often use semicolons or parentheses.

Python relies on indentation, using whitespace, to define scope; such as the scope of loops, functions and classes. Other programming languages often use curly-brackets for this purpose.

Python is Interpreted

Many languages are compiled, meaning the source code you create needs to be translated into machine code, the language of your computer's processor, before it can be run. Programs written in an interpreted language are passed straight to an interpreter that runs them directly. This makes for a quicker development cycle because you just type in your code and run it, without the intermediate compilation step.

One potential downside to interpreted languages is execution speed. Programs that are compiled into the native language of the computer processor tend to run more quickly than interpreted programs. For some applications that are particularly computationally intensive, like graphics processing or intense number crunching, this can be limiting.

In practice, however, for most programs, the difference in execution speed is measured in milliseconds, or seconds at most, and not appreciably noticeable to a human user. The expediency of coding in an interpreted language is typically worth it for most applications.

For all its syntactical simplicity, Python supports most constructs that would be expected in a very high-level language, including complex dynamic data types, structured and functional programming, and object-oriented programming.

Additionally, a very extensive library of classes and functions is available that provides capability well beyond what is built into the language, such as database manipulation or GUI programming.

Python accomplishes what many programming languages don't: the language itself is simply designed, but it is very versatile in terms of what you can accomplish with it.

Introduction to Machine Learning using Python

Machine learning is a type of artificial intelligence (AI) that provides computers with the ability to learn without being explicitly programmed. Machine learning focuses on the development of Computer Programs that can change when exposed to new data. In this article, we'll see basics of Machine Learning, and implementation of a simple machine learning algorithm using python.

Machine learning algorithms use statistical techniques to enable machines to improve their performance on a task through experience. These algorithms are widely used in applications such as recommendation systems, image recognition, natural language processing, and predictive analytics. By the end of this article, you'll have a foundational understanding of how machine learning works and how to apply it to real-world problems using Python.

It's important to understand the different types of machine learning:

supervised, unsupervised, and reinforcement learning.

Each type has its own use cases and is suited for solving different kinds of problems. We'll also explore the basic steps involved in building a machine learning model, such as data preprocessing, model training, and evaluation.

3.3.2 Convolutional Neural Networks (CNN) Introduction

Convolutional neural networks (CNN) sounds like a weird combination of biology and math with a little CS sprinkled in, but these networks have been some of the most influential innovations in the field of computer vision. 2012 was the first year that neural nets grew to prominence as Alex Krizhevsky used them to win that year's ImageNet competition (basically, the annual Olympics of computer vision), dropping the classification error record from 26% to 15%, an astounding improvement at the time. Ever since then, a host of companies have been using deep learning at the core of their services. Facebook uses neural nets for their automatic tagging algorithms, Google for their photo search, Amazon for their product recommendations, Pinterest for their home feed personalization, and Instagram for their search infrastructure.

Inputs and Outputs

When a computer sees an image (takes an image as input), it will see an array of pixel values. Depending on the resolution and size of the image, it will see a 32 x 32 x 3 array of numbers (The 3 refers to RGB values). Just to drive home the point, let's say we have a color image in JPG form and its size is 480 x 480. The representative array will be 480 x 480 x 3. Each of these numbers is given a value from 0 to 255 which describe the pixel intensity at that point. These numbers, while meaningless to us when we perform image classification, are the only inputs available to the computer. The idea is that you give the computer this array of numbers and it will output numbers that describe the probability of the image being a certain class (.80 for cat, .15 for dog, .05 for bird, etc).

What We Want the Computer to do

Now that we know the problem as well as the inputs and outputs, let's think about how to approach this. What we want the computer to do is to be able to differentiate between all the images it's given and figure out the unique features that make a dog a dog or that make a cat a cat. This is the process that goes on in our minds subconsciously as well. When we look at a picture of a dog, we can classify it as such if the picture has identifiable features such as paws or 4 legs. In a similar way, the computer is able perform image classification by looking for low level features such as edges and curves, and then building up to more abstract concepts through a series of convolutional layers. This is a general overview of what a CNN does.

Biological Connection

But first, a little background. When you first heard of the term convolutional neural networks, you may have thought of something related to neuroscience or biology, and you would be right. Sort of. CNNs do take a biological inspiration from the visual cortex. The visual cortex has small regions of cells that are sensitive to specific regions of the visual field. This idea was expanded upon by a fascinating experiment by Hubel and Wiesel in 1962 (Video) where they showed that some individual neuronal cells in the brainresponded (or fired) only in the presence of edges of a certain orientation. For example, some neurons fired when exposed to vertical edges and some when shown horizontal or diagonal edges. Hubel and Wiesel found out that all of these neurons were organized in a columnar architecture and that together, they were able to produce visual perception. This idea of specialized components inside of a system having specific tasks (the neuronal cells in the visual cortex looking for specific characteristics) is one that machines use as well, and is the basis behind CNNs.

The Problem Space

Image classification is the task of taking an input image and outputting a class (a cat, dog, etc) or a probability of classes that best describes the image. For humans, this task of recognition is one of the first skills we learn from the moment we are born and is one that comes naturally and effortlessly as adults. Without even thinking twice, we're able to quickly and seamlessly identify the environment we are in as well as the objects that surround us. When we see an image or just when we look at the world around us, most of the time we are able to immediately characterize the scene and give each object a label, all without even consciously noticing. These skills of being able to quickly recognize patterns, generalize from prior knowledge, and adapt to different image environments are ones that we do not share with our fellow machines. For machines, replicating this level of perception is a complex challenge that requires large amounts of labeled data, powerful computational resources, and advanced algorithms. Image classification tasks often rely on deep learning models, particularly Convolutional Neural Networks (CNNs), which are designed to mimic the way the human brain processes visual information. Through training on massive datasets, these models learn to detect edges, textures, shapes, and eventually high-level features that allow them to differentiate between different image classes. While machines have made impressive strides in this field, achieving human-level accuracy remains a major area of ongoing research and innovation.

3.3.3 Convolution Neural Network Algorithm

Convolutional Neural Network (CNN) algorithm is a multilayer perceptron specially designed for the identification and processing of two-dimensional image information. It typically consists of multiple layers: an input layer, convolution layers, sampling (pooling) layers, and an output layer. In deeper network architectures, the convolution and sampling layers are often stacked multiple times to extract increasingly abstract features. Unlike Restricted Boltzmann Machines (RBMs), CNNs do not require every neuron to be connected to all neurons in adjacent layers. Instead, they use local connections, where each neuron focuses only on a small region of the input image — known as the local receptive field.

A key feature of CNNs is weight sharing: all neurons within the same feature map share the same convolution kernel, which dramatically reduces the number of trainable parameters and helps prevent overfitting. CNNs follow two main processes: convolution and sampling (pooling). During the convolution process, a trainable filter (Fx) scans across the input image to compute dot products, generating a feature map. A bias (bx) is added, and the output forms the convolution layer (Cx). In the sampling process, a neighbourhood of pixels is reduced to a single value using operations like max pooling or average pooling. This result is then weighted by scalar weights (Wx+1), added with a bias (bx+1), and passed through a non-linear activation function, producing a down sampled feature map (Sx+1).

The core technologies behind CNNs include local receptive fields, weight sharing, and subsampling, which together enable automatic feature extraction while significantly reducing the number of parameters. This makes CNNs highly effective for image recognition and classification tasks. The algorithm's ability to learn spatial hierarchies of features allows it to be robust to variations like scaling, translation, and slight distortions. Additionally, CNNs align well with the structure of image and speech data, making them particularly powerful in fields like image classification, object detection, face recognition, and speech recognition.

With the rise of computational power and the availability of large labeled datasets, CNNs have become the backbone of modern computer vision systems. Their scalability, efficiency, and performance continue to drive breakthroughs in AI applications ranging from autonomous driving to medical imaging.

A convolutional neural network (CNN) is a deep friendly neural network designed from biologically controlled modules. They are applied most frequently used for image processing. CNN's work differently in that they treat data as spatial. Instead of being connected to all the neurons in the previous layer, the neurons are connected only to neighbouring neurons and all have the same weight. This simplification in connections means that the network maintains the spatial aspect of the data set. The word convolution refers to the filtering process that occurs in this type of network. If an image is complex, the convolutional neural network simplifies it, so it can be better elaborated and understood. Like a normal neural network, CNN is also composed of several levels. There are a couple of levels that make the convolutional level and the grouping level unique. However, like other neural networks, it will also have a rectified linear unit layer and a fully connected layer. The ReLU level acts as an activation function, ensuring non-linearity as the data moves through each level of the network.

Without it, the data entered in each level would lose the dimensionality that we want to maintain. Meanwhile, the fully connected layer allows you to classify the data set.

However, the classic, and arguably most popular, use case of these networks is for image processing. Within image processing, let's take a look at how to use these CNNs for image classification.

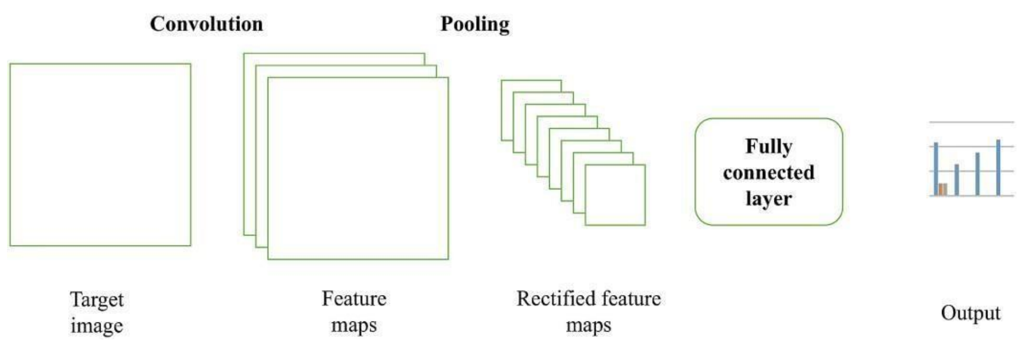


Fig 3.1: Steps in CNN

First Layer – Math Part

The first layer in a CNN is always a Convolutional Layer. First thing to make sure you remember is what the input to this conv (I'll be using that abbreviation a lot) layer is. Like we mentioned before, the input is a 32 x 32 x 3 array of pixel values. Now, the best way to explain a conv layer is to imagine a flashlight that is shining over the top left of the image. Let’s say that the light this flashlight shines covers a 5 x 5 area. And now, let’s imagine this flashlight sliding across all the areas of the input image. In machine learning terms, this flashlight is called a filter (or sometimes referred to as a neuron or a kernel) and the region that it is shining over is called the receptive field. Now this filter is also an array of numbers (the numbers are called weights or parameters). A very important note is that the depth of this filter has to be the same as the depth of the input (this makes sure that the math works out), so the dimension of this filter is 5 x 5 x 3. Now, let’s take the first position the filter is in for example. It would be the top left corner. As the filter is sliding, or convolving, around the input image, it is multiplying the values in the filter with the original pixel values of the image (aka computing element wise multiplications). These multiplications are all summed up (mathematically speaking, this would be 75 multiplications in total). So now you have a single number. Remember, this number is just representative of when the filter is at the top left of the image. Now, we repeat this process for every location on the input volume. (Next step would be moving the filter to the right by 1 unit, then right again by 1, and so on). Every unique location on the input volume produces a number. After sliding the filter over all the locations, you will find out that what you’re left with is a 28 x 28 x 1 array of numbers, which we call an activation map or feature map. The reason you get a 28 x 28 array is that there are 784 different locations that a 5 x 5 filter can fit on a 32 x 32 input image. These 784 numbers are mapped to a 28 x 28 array.

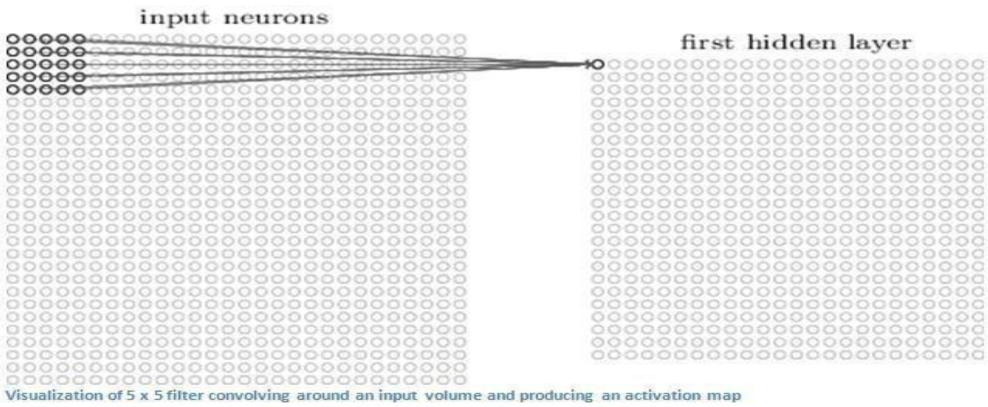


Fig 3.2: First Neuron Layer

Let’s say now we use two 5 x 5 x 3 filters instead of one. Then our output volume would be 28 x 28 x 2. By using more filters, we are able to preserve the spatial dimensions better. Mathematically, this is what’s going on in a convolutional layer.

First Layer – High Level Perspective

However, let’s talk about what this convolution is actually doing from a high level. Each of these filters can be thought of as feature identifiers. When I say features, I’m talking about things like straight edges, simple colors, and curves. Think about the simplest characteristics that all images have in common with each other. Let’s say our first filter is 7 x 7 x 3 and is going to be a curve detector. (In this section, let’s ignore the fact that the filter is 3 units deep and only consider the top depth slice of the filter and the image, for simplicity.)As a curve detector, the filter will have a pixel structure in which there will be higher numerical values along the area that is a shape of a curve (Remember, these filters that we’re talking about as just numbers!).

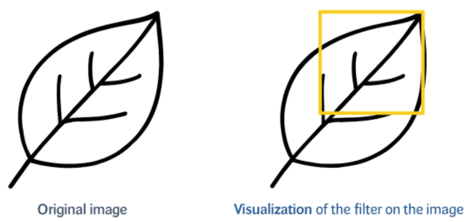


Fig 3.3: Original Image

0	0	0	0	0	30	0
0	0	0	0	30	0	0
0	0	0	30	0	0	0
0	0	0	30	0	0	0
0	0	0	30	0	0	0
0	0	0	30	0	0	0
0	0	0	0	0	0	0

Pixel representation of filter

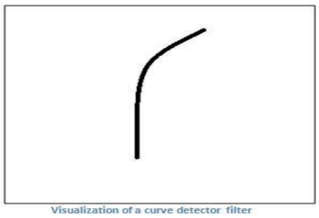


Fig 3.4: Visualization of a curve detector filter

Now, let's go back to visualizing this mathematically. When we have this filter at the top left corner of the input volume, it is computing multiplications between the filter and pixel values at that region. Now let's take an example of an image that we want to classify, and let's put our filter at the top left corner.

Remember, what we have to do is multiply the values in the filter with the original pixel values of the image.

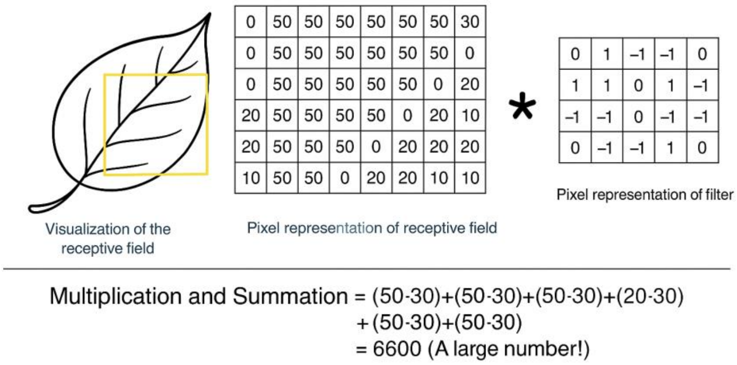


Fig 3.5: Visualization of the respective field

Basically, in the input image, if there is a shape that generally resembles the curve that this filter is representing, then all of the multiplications summed together will result in a large value! Now let's see what happens when we move our filter.

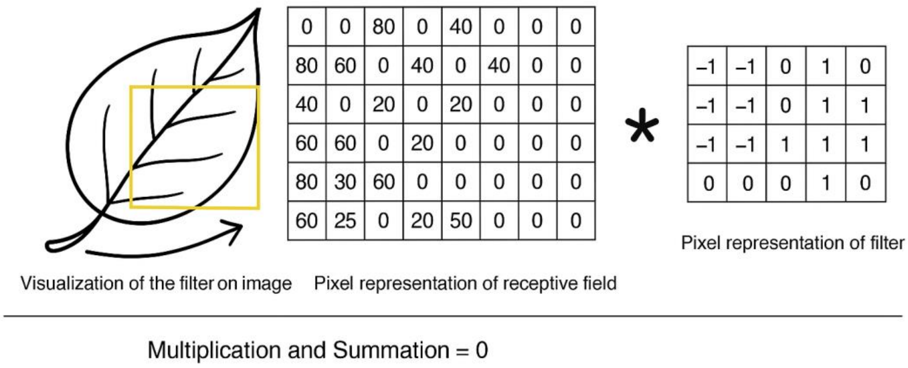


Fig 3.6: Pixel Representation of Filter

The value is much lower! This is because there wasn't anything in the image section that responded to the curve detector filter. Remember, the output of this conv layer is an activation map. So, in the simple case of a one filter convolution. curve detector, the activation map will show the areas in which there at mostly likely to be curves in the picture. In this example, the top left value of our 26 x 26 x 1 activation map (26 because of the 7x7 filter instead of 5x5) will be 6600. This high value means that it is likely that there is some sort of curve in the input volume that caused the filter to activate. The top right value in our activation map will be 0 because there wasn't anything in the input volume that caused the filter to activate (or more simply said, there wasn't a curve in that region of the original image). Remember, this is just for one filter. This is just a filter that is going to detect lines that curve outward and to the right. We can have other filters for lines that curve to the left or for straight edges. The more filters, the greater the depth of the activation map, and the more information we have about the input volume.

Going Deeper Through the Network

Now in traditional convolutional neural network architecture, there are other layers that are interspersed between these conv layers. I'd strongly encourage those interested to read up on them and understand their function and effects, but in a general sense, they provide nonlinearities and preservation of dimension that help to improve the robustness of the network and control over fitting. A classic CNN architecture would look like this.

Input -> Conv -> ReLU -> Conv -> ReLU -> Pool -> ReLU -> Conv -> ReLU -> Pool ->Fully Connected

The last layer, however, is an important one and one that we will go into later on. Let's just take a step back and review what we've learned so far. We talked about what the filters in the first conv layer are designed to detect. They detect low level features such as edges and curves. As one would imagine, in order predicting whether an image is a type of object, we need the network to be able to recognize higher level features such as hands or paws or ears. So let's think about what the output of the network is after the first conv layer. It would be a 28 x 28 x 3 volume (assuming we use three 5 x 5 x 3 filters). When we go through another conv layer, the output of the first conv layer becomes the input of the 2^{nd} conv layer. Now, this is a little bit harder to visualize. When we were talking about the first layer, the input was just the original image. However, when we're talking about the 2^{nd} conv layer, the input is the activation map(s) that result from the first layer. So each layer of the input is basically describing the locations in the original image for where certain low level features appear. Now when you apply a set of filters on top of that (pass it through the 2^{nd} conv layer), the output will be activations that represent higher level features. Types of these features could be semicircles (combination of a curve and straight edge) or squares (combination of several straight edges). As you go through the network and go through more conv layers, you get activation maps that represent more and more complex features. By the end of the network, you may have some filters that activate when there is handwriting in the image, filters that activate when they see pink objects, etc. If you want more information about visualizing filters in ConvNets, Matt Zeiler and Rob Fergus had an excellent research paper discussing the topic. Jason Yosinski also has a video on YouTube that provides a great visual representation. Another interesting thing to note is that as you go deeper into the network, the filters begin to have a larger and larger receptive field, which means that they are able to consider information from a larger area of the original input volume (another way of putting it is that they are more responsive to a larger region of pixel space).

The convolution layer computes the output of neurons that are connected to local regions or receptive fields in the input, each computing a dot product between their weights and a small receptive field to which they are connected to in the input volume. Each computation leads to extraction of a feature map from the input image. In other words, imagine you have an image represented as a 5x5 matrix of values, and you take a 3x3 matrix and slide that 3x3 window or kernel around the image. At each position of that matrix, you multiply the values of your 3x3 window by the values in the image that are currently being covered by the window. As a result, you'll get a single number that represents all the values in that window of the images. You use this layer to filtering: as the window moves over the image, you check for patterns in that section of the image. This works because of filters, which are multiplied by the values outputted by the convolution.

The objective of subsampling is to get an input representation by reducing its dimensions, which helps in reducing overfitting. One of the techniques of subsampling is max pooling. With this technique, you select the highest pixel value from a region depending on its size. In other words, max pooling takes the largest value from the window of the image currently covered by the kernel. For example, you can have a max-pooling layer of size 2 x 2 will select the maximum pixel intensity value from 2 x 2 region. You're right to think that the pooling layer then works a lot like the convolution layer! You also take a kernel or a window and move it over the image; The only difference is the function that is applied to the kernel and the image window isn't linear.

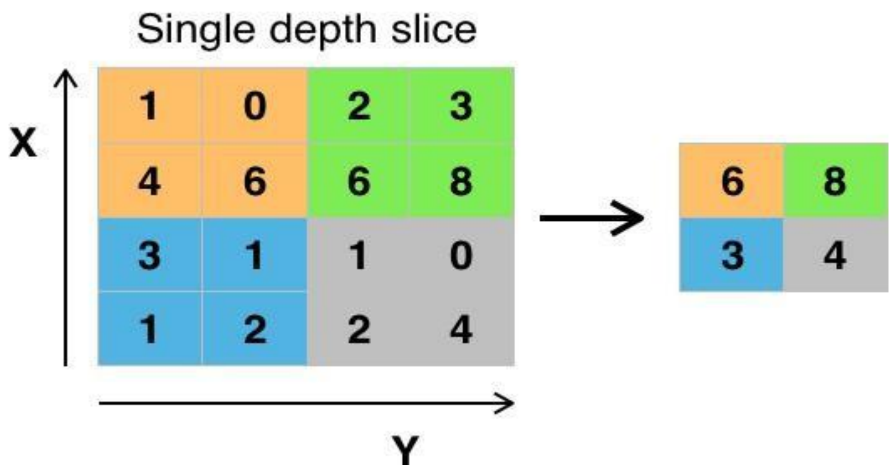


Fig 3.7: Max Pooling

The objective of the fully connected layer is to flatten the high-level features that are learned by convolutional layers and combining all the features. It passes the flattened output to the output layer where you use a softmax classifier or a sigmoid to predict the input class label.

How Companies Use CNNs

Data, data, data. The companies that have lots of this magic 4 letter word are the ones that have an inherent advantage over the rest of the competition. The more training data that you can give to a network, the more training iterations you can make, the more weight updates you can make, and the better tuned to the network is when it goes to production. Facebook (and Instagram) can use all the photos of the billion users it currently has, Pinterest can use information of the 50 billion pins that are on its site, Google can use search data, and Amazon can use data from the millions of products that are bought every day. And now you know the magic behind how they use it.

3.4 APPLICATIONS OF CNN

Image recognition

CNNs are often used in image recognition systems. In 2012 an error rate of
0.23 percent on the MNIST database was reported. Another paper on using CNN for image classification reported that the learning process was "surprisingly fast"; in the same paper, the best published results as of 2011 were achieved in the

MNIST database and the NORB database. Subsequently, a similar CNN called AlexNet[74] won the ImageNet Large Scale Visual Recognition Challenge 2012.

When applied to facial recognition, CNNs achieved a large decrease in error rate. Another paper reported a 97.6 percent recognition rate on "5,600 still images of more than 10 subjects". CNNs were used to assess video quality in an objective way after manual training; the resulting system had a very low root mean square error.

The ImageNet Large Scale Visual Recognition Challenge is a benchmark in object classification and detection, with millions of images and hundreds of object classes. In the ILSVRC 2014,[76] a large-scale visual recognition challenge, almost every highly ranked team used CNN as their basic framework. The winner GoogLeNet(the foundation of DeepDream) increased the mean average precision of object detection to 0.439329, and reduced classification error to 0.06656, the best result to date. Its network applied more than 30 layers. That performance of convolutional neural networks on the ImageNet tests was close to that of humans. The best algorithms still struggle with objects that are small or thin, such as a small ant on a stem of a flower or a person holding a quill in their hand. They also have trouble with images that have been distorted with filters, an increasingly common phenomenon with modern digital cameras. By contrast, those kinds of images rarely trouble humans. Humans, however, tend to have trouble with other issues. For example, they are not good at classifying objects into fine-grained categories such as the particular breed of dog or species of bird, whereas convolutional neural networks handle this.[citation needed]

In 2015 a many-layered CNN demonstrated the ability to spot faces from a wide range of angles, including upside down, even when partially occluded, with competitive performance. The network was trained on a database of 200,000 images that included faces at various angles and orientations and a further 20 million images without faces. They used batches of 128 images over 50,000 iterations.

Video analysis

Compared to image data domains, there is relatively little work on applying CNNs to video classification. Video is more complex than images since it has another (temporal) dimension. However, some extensions of CNNs into the video domain have been explored. One approach is to treat space and time as equivalent dimensions of the input and perform convolutions in both time andspace. Another way is to fuse the features of two convolutional neural networks, one for the spatial and one for the temporal stream.Long short-term memory (LSTM) recurrent units are typically incorporated after the CNN to account for inter- frame or inter-clip dependencies. Unsupervised learning schemes for training spatio- temporal features have been introduced, based on Convolutional GatedRestricted Boltzmann Machines and Independent Subspace Analysis.

Natural language processing

CNNs have also been explored for natural language processing. CNN models are effective for various NLP problems and achieved excellent results in semantic parsing, search query retrieval, sentence modeling, classification, prediction and other traditional NLP tasks.

Anomaly Detection

A CNN with 1-D convolutions was used on time series in the frequency domain (spectral residual) by an unsupervised model to detect anomalies in the time domain.

Drug discovery

CNNs have been used in drug discovery. Predicting the interaction between molecules and biological proteins can identify potential treatments. In 2015, Atomwise introduced AtomNet, the first deep learning neural network for structure-based rational drug design. The system trains directly on 3-dimensional representations of chemical interactions. Similar to how image recognition networks learn to compose smaller, spatially proximate features into larger, complex structures, Atom Net discovers chemical features, such as aromaticity, sp3 carbons and hydrogen bonding. Subsequently, Atom Net was used to predict novel candidate biomolecules for multiple disease targets, most notably treatments for the Ebola virus and multiple sclerosis.

Health risk assessment and biomarkers of aging discovery

CNNs can be naturally tailored to analyze a sufficiently large collection of time series data representing one-week-long human physical activity streams augmented by the rich clinical data (including the death register, as provided by, e.g., the NHANES study). A simple CNN was combined with Cox-Gompertz proportional hazards model and used to produce a proof-of-concept example of digital biomarkers of aging in the form of all-causes-mortality
Predict.

Checkers game

CNNs have been used in the game of checkers. From 1999 to 2001, Fogel and Chellapilla published papers showing how a convolutional neural network could learn to play checker using co-evolution. The learning process did not use prior human professional games, but rather focused on a minimal set of information contained in the checkerboard: the location and type of pieces, and the difference in number of pieces between the two sides. Ultimately, the program (Blondie24) was tested on 165 games against players and ranked in the highest 0.4%. It also earned a win against the program Chinook at its "expert" level of play.

Go

CNNs have been used in computer Go. In December 2014, Clark and
Storkey published a paper showing that a CNN trained by supervised learning from a database of human professional games could outperform GNU Go and win some games against Monte Carlo tree search Fuego in a fraction of the time it took Fuego to play. Later it was announced that a large 12-layer convolutional neural network had correctly predicted the professional move in 55% of positions, equaling the accuracy of a 6 dan human player. When the trained convolutional network was used directly to play games of Go, without any search, it beat the traditional search program GNU Go in 97% of games, and matched the performance of the Monte Carlo tree search program Fuego simulating ten thousand playouts (about a million positions) per move.

A couple of CNNs for choosing moves to try ("policy network") and evaluating positions ("value network") driving MCTS were used by AlphaGo, the first to beat the best human player at the time.

Time series forecasting

Recurrent neural networks are generally considered the best neural network architectures for time series forecasting (and sequence modeling in general), but recent studies show that convolutional networks can perform comparably or even better. Dilated convolutions might enable one-dimensional convolutional neural networks to effectively learn time series dependences. Convolutions can be implemented more efficiently than RNN-based solutions, and they do not suffer from vanishing (or exploding) gradients. Convolutional networks can provide an improved forecasting performance when there are multiple similar time series to learn from. CNNs can also be applied to further tasks in time series analysis.

Fine-tuning

For many applications, the training data is less available. Convolutional neural networks usually require a large amount of training data in order to avoid overfitting. A common technique is to train the network on a larger data set from a related domain. Once the network parameters have converged an additional training step is performed using the in-domain data to fine-tune the network weights. This allows convolutional networks to be successfully applied to problems with small training sets.

Human interpretable explanations

End-to-end training and prediction are common practice in computer vision. However, human interpretable explanations are required for critical systems such as a self-driving cars. With recent advances in visual salience, spatial and temporal attention, the most critical spatial regions/temporal instants could be visualized to justify the CNN predictions.

3.5 System Architecture

The initial step is to delete the RGB channel data from the information image to distinguish the implant data from the anterior area. To do this, data is extracted from the excess green (ExG) and excess red (Exp) channels.

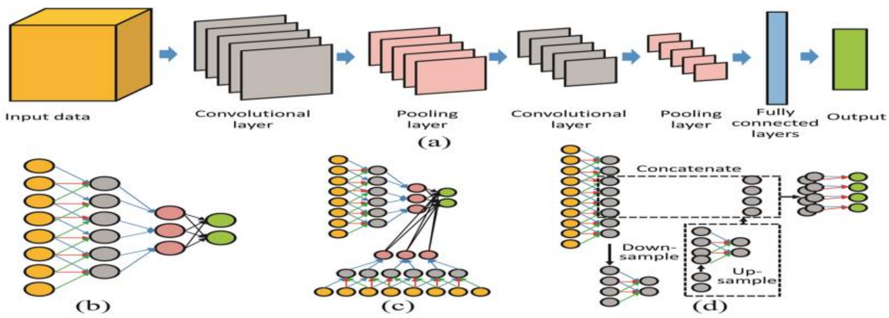


Fig 3.8: Outline of the Proposed System

Then, the overabundance data is removed from the abundance of green data to progressively contain the green channel data. When the double limit is applied in the improved vegetative archive, the data, a double intrigue district is divided into plants. The next advance is to use the first double image shadow pixel as a cover that separates the leaves as secondary images. The division of the test sheet images into the dataset using the vegetative record mentioned above. The important component in the representation of an image is shading. The shading minutes are obtained through the highlighted facts, for example the average and the standard deviation evaluated in the three RGB channels (red, green, blue). The main measures for requesting an image are acquired from the standard deviation and the average.

The surface of the image can be estimated using the concurrency pane, the different shading elements or the grayscale estimates of the image. The highlights that are created using this strategy are known as Haralick highlights. A calculated regression classifier is used to distinguish the restoration plan. Since our proposed strategy consists of the characterization of numerous classes, the relapse model calculated in parallel has been extended to the calculated multinomial relapse. Through a combination of relapses calculated in pairs, the multinomial logit thinks of different encounters. This allows you to compare each brand class with a reference classification in the data set.

3.6 UML DIAGRAMS

3.6.1 Use Case Diagram

A use case diagram in the Unified Modeling Language (UML) is a type of behavioral diagram defined by and created from a Use-case analysis. Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goals (represented as use cases), and any dependencies between those use cases. The main purpose of a use case diagram is to show what system functions are performed for which actor. Roles of the actors in the system can be depicted.

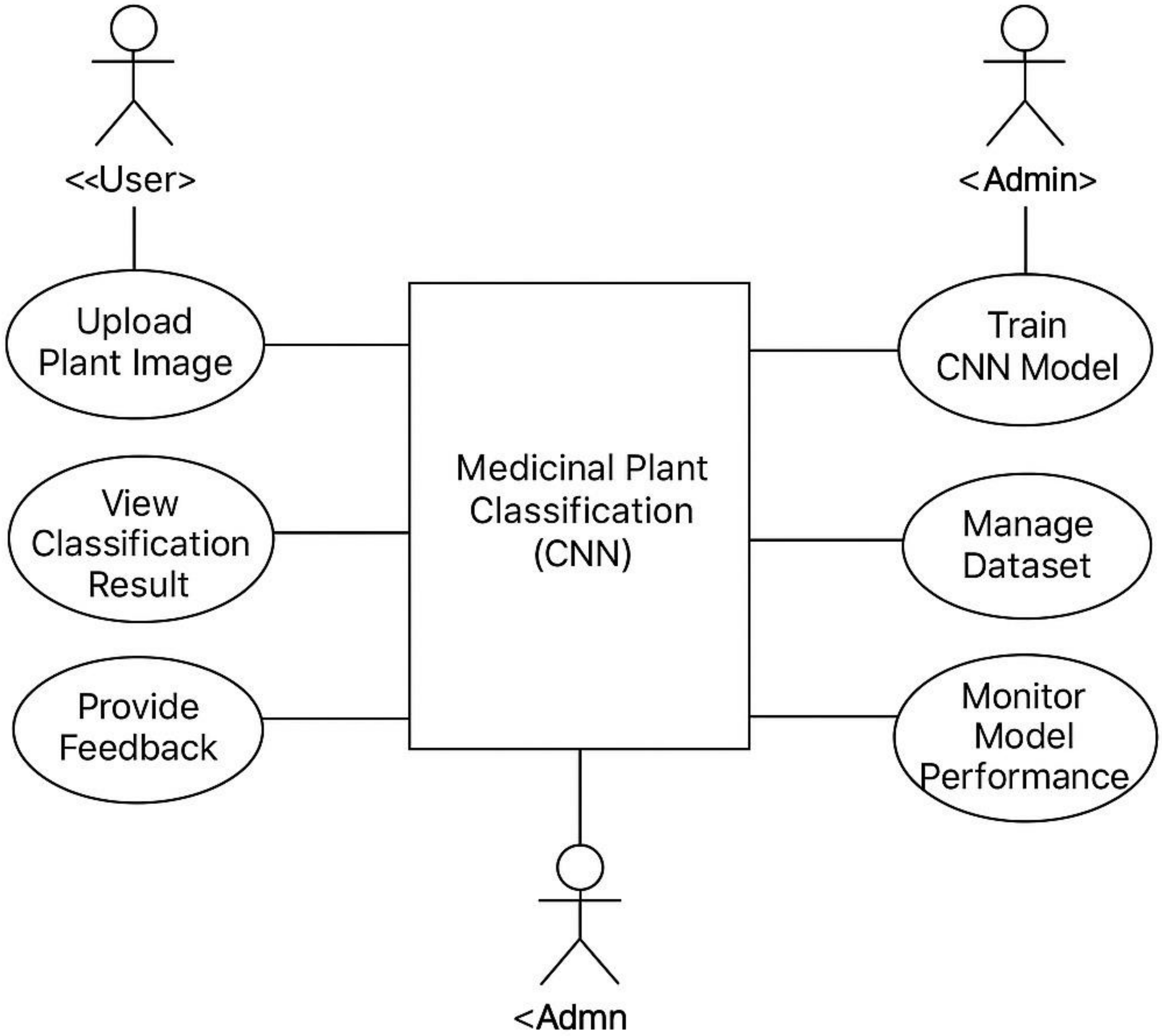


Fig 3.9:- Use Case Diagram

3.6.2 Class Diagram

In software engineering, a class diagram in the Unified Modelling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among the classes. It explains which class contains information.

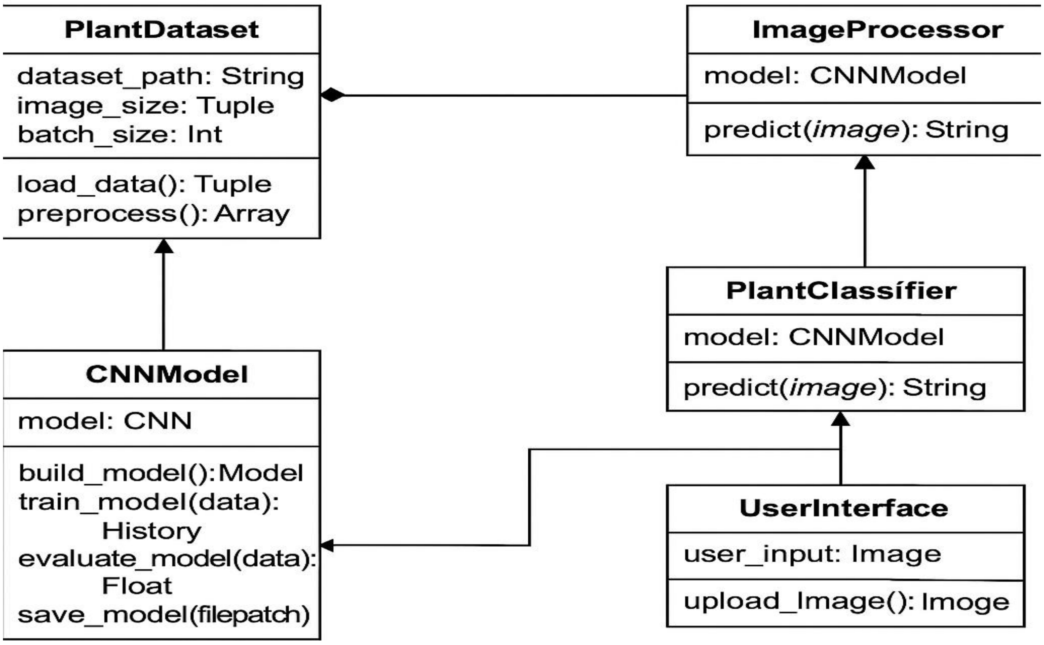


Fig 3.10: Class Diagram

3.6.3 Sequence Diagram

A sequence diagram in Unified Modelling Language (UML) is a kind of interaction diagram that shows how processes operate with one another and in what order. It is a construct of a Message Sequence Chart. Sequence diagrams are sometimes called event diagrams, event scenarios, and timing diagrams.

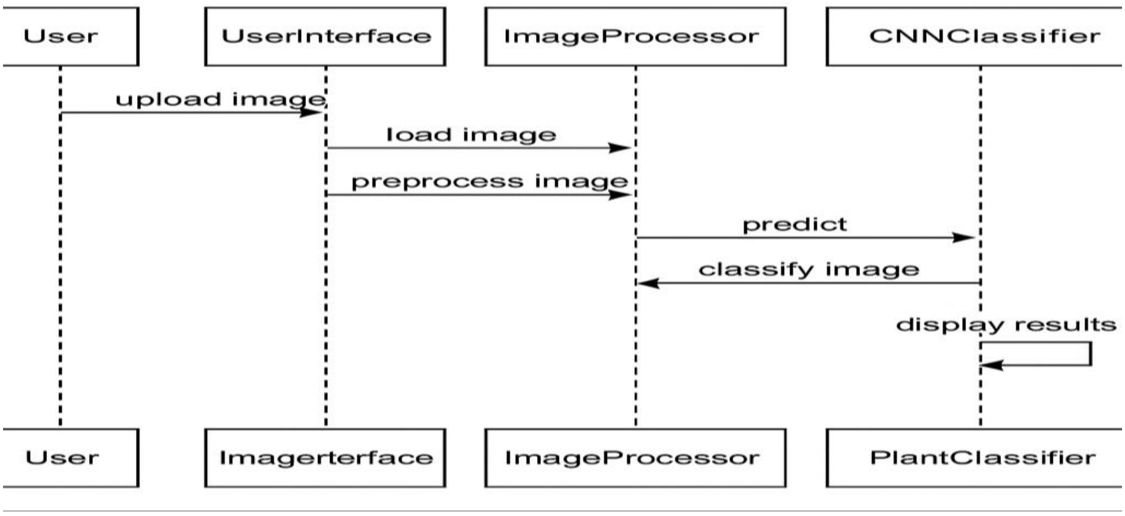


Fig 3.11:- Sequence Diagram

3.6.4 Collaboration Diagram

In collaboration diagram the method call sequence is indicated by some numbering technique as shown below. The number indicates how the methods are called one after another. We have taken the same order management system to describe the collaboration diagram. The method calls are similar to that of a sequence diagram. But the difference is that the sequence diagram does not describe the object organization whereas the collaboration diagram shows the object organization.

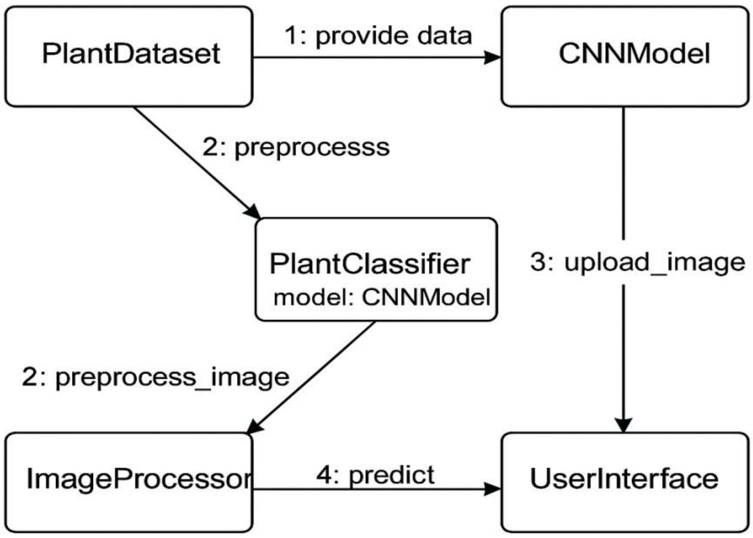


Fig 3.12: Collaboration Diagram

3.6.5 Deployment Diagram

Deployment diagram represents the deployment view of a system. It is related to the component diagram. Because the components are deployed using the deployment diagrams. A deployment diagram consists of nodes. Nodes are nothing but physical hardware’s used to deploy the application.

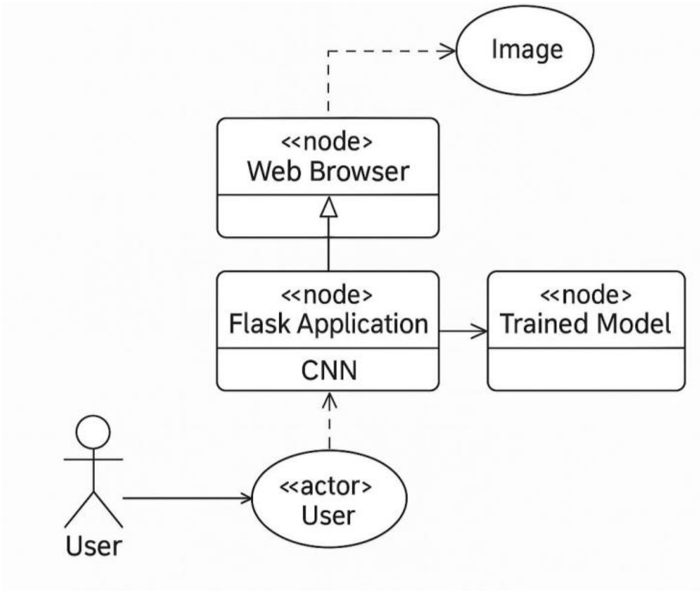


Fig 3.13:- Deployment Diagram

3.6.6 Activity Diagram

Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency. In the Unified Modelling Language, activity diagrams can be used to describe the business and operational step by-step workflows of components in a system. An activity diagram shows the overall flow of control.

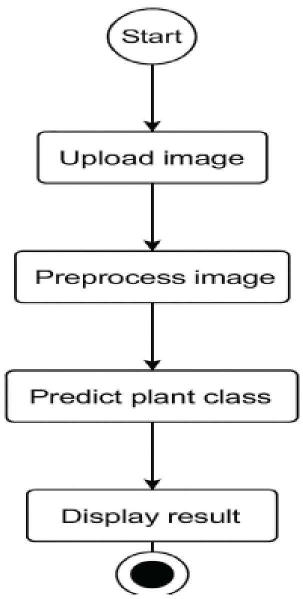


Fig 3.14:- Activity Diagram

3.6.7 Component Diagram

A component diagram, also known as a UML component diagram, describes the organization and wiring of the physical components in a system. Component diagrams are often drawn to help model implementation details and double-check that every aspect of the system's required functions is covered by planned development.

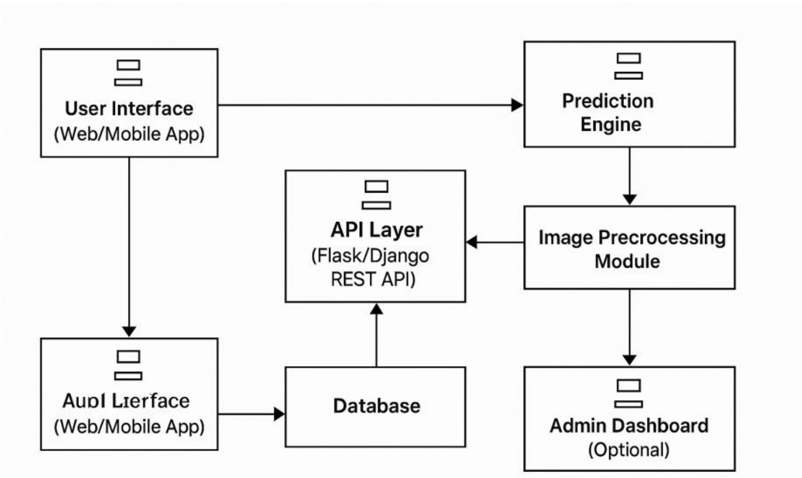


Fig 3.15:- Component Diagram

3.7 ER Diagram

An Entity–relationship model (ER model) describes the structure of a database with the help of a diagram, which is known as Entity Relationship Diagram (ER Diagram). An ER model is a design or blueprint of a database that can later be implemented as a database. The main components of E-R model are: entity set and relationship set. An ER diagram shows the relationship among entity sets. An entity set is a group of similar entities and these entities can have attributes. In terms of DBMS, an entity is a table or attribute of a table in database, so by showing relationship among tables and their attributes, ER diagram shows the complete logical structure of a database. Let's have a look at a simple ER diagram to understand this concept.

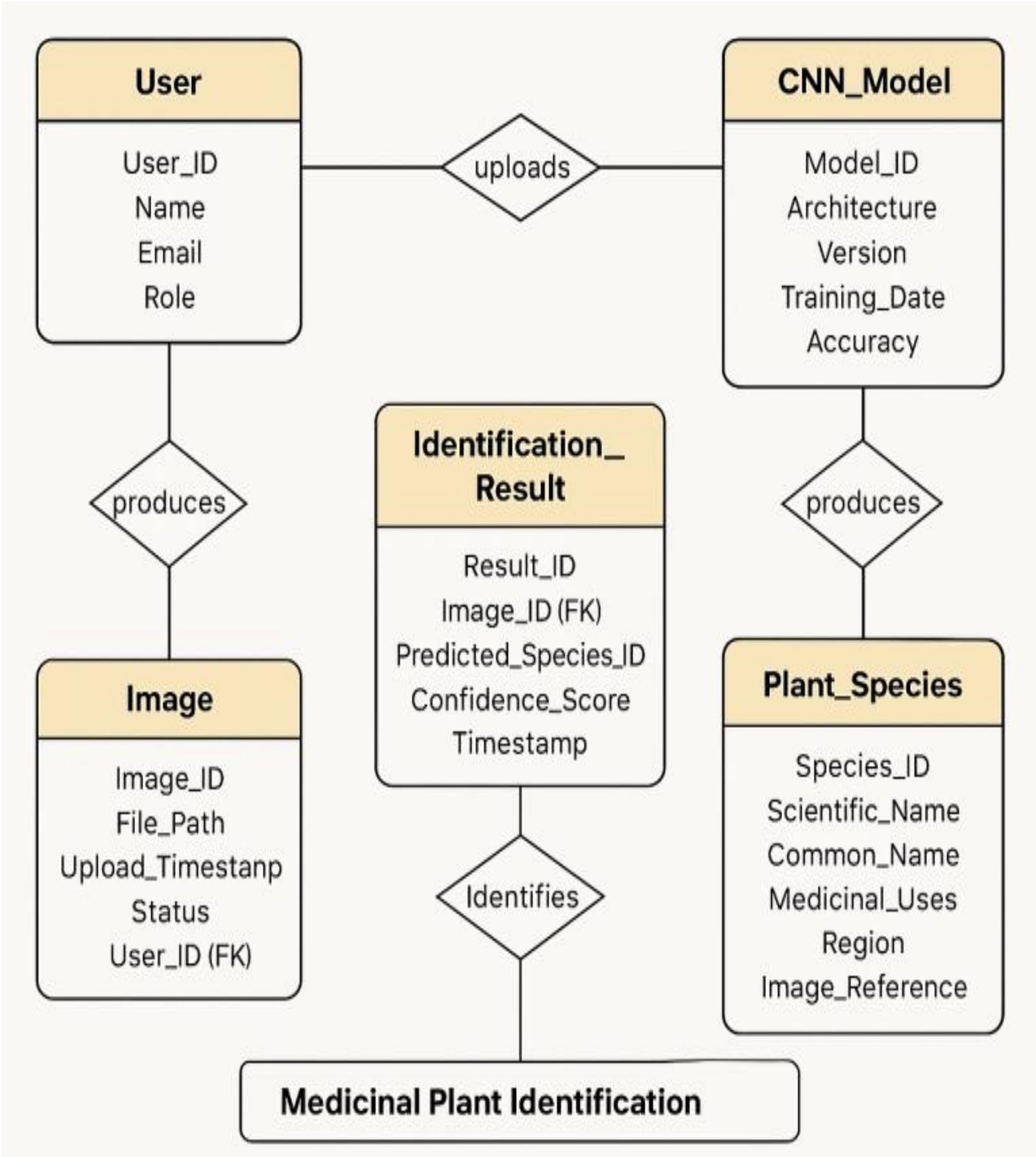


Fig 3.16: ER Diagram

CHAPTER 4
MODULE DESCRIPTION

4.1 System Study

Feasibility Study

The feasibility of the project is analyzed in this phase and business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the company. For feasibility analysis, some understanding of the major requirements for the system is essential.

Three key considerations involved in the feasibility analysis are

Economical Feasibility

Technical Feasibility

Social Feasibility

Legal Feasibility

Operational Feasibility

Scheduling Feasibility

Economical Feasibility

This study is carried out to check the economic impact that the system will have on the organization. The amount of fund that the company can pour into the research and development of the system is limited. The expenditures must be justified. Thus the developed system as well within the budget and this was achieved because most of the technologies used are freely available. Only the customized products had to be purchased.

Technical Feasibility

This study is carried out to check the technical feasibility, that is, the technical requirements of the system. Any system developed must not have a high demand on the available technical resources. This will lead to high demands on the available technical resources. This will lead to high demands being placed on the client. The developed system must have a modest requirement, as only minimal or null changes are required for implementing this system.

Social Feasibility

The aspect of study is to check the level of acceptance of the system by the user. This includes the process of training the user to use the system efficiently. The user must not feel threatened by the system, instead must accept it as a necessity. The level of acceptance by the users solely depends on the methods that are employed to educate the user about the system and to make him familiar with it. His level of confidence must be raised so that he is also able to make some constructive criticism, which is welcomed, as he is the final user of the system.

Legal Feasibility

This assessment investigates whether any aspect of the proposed project conflicts with legal requirements like zoning laws, data protection acts or social media laws. Let's say an organization wants to construct a new office building in a specific location. A feasibility study might reveal the organization's ideal location isn't zoned for that type of business. That organization has just saved considerable time and effort by learning that their project was not feasible right from the beginning.

Operational Feasibility

This assessment involves undertaking a study to analyze and determine whether— and how well—the organization's needs can be met by completing the project. Operational feasibility studies also examine how a project plan satisfies the requirements identified in the requirements analysis phase of system development.

Scheduling Feasibility

This assessment is the most important for project success; after all, a project will fail if not completed on time. In scheduling feasibility, an organization estimates how much time the project will take to complete.

When these areas have all been examined, the feasibility analysis helps identify any constraints the proposed project may face, including:

Internal Project Constraints: Technical, Technology, Budget, Resource, etc.

Internal Corporate Constraints: Financial, Marketing, Export, etc.

External Constraints: Logistics, Environment, Laws, and Regulations.

4.2 SYSTEM TESTING

Testing ensures the accuracy, reliability, and usability of the system. Various testing techniques were employed to examine the functionality of each module and the system as a whole.

4.2.1 Unit Testing

Each function and module was tested independently to ensure that it worked as intended. For instance, separate tests were run for pre-processing images, loading models, and handling prediction output. This ensures that even the smallest components are error-free and reliable.

4.2.2 Integration Testing

After unit testing, modules were integrated and tested collectively. This phase ensured that data passed from one module to another without issues, such as passing pre-processed images into the model and rendering predictions on the frontend. Integration testing ensured harmony across all interconnected components.

4.2.3 Acceptance Testing

The completed system was shared with a sample group of intended users, including students, researchers, and medical professionals. Their feedback was overwhelmingly positive, particularly praising the system's speed, ease of use, and diagnostic accuracy. Minor usability improvements were made based on their input.

4.2.4 Functional Testing

Each feature of the system was tested in a real-world scenario. For instance, uploading a brain MRI image was tested to ensure proper handling of both valid and invalid inputs. All buttons, links, and input forms were verified for correct operation. The results module was tested to ensure correct label rendering.

4.2.5 White Box Testing

This testing technique was used to evaluate the internal logic, functions, and flow of the code. It ensured that the model was initialized properly, weights were being loaded from the correct file, and that prediction logic was functioning according to the architecture's expectations. Python scripts were debugged and profiled to optimize runtime performance.

4.2.6 Black Box Testing

This test focused on the application's behavior without knowing the internal code. It simulated end-user scenarios such as uploading a corrupted file or using a non-image format to verify the system's robustness. The software was found to gracefully handle all invalid inputs, showing appropriate error messages without crashing.

4.3 Module Description

4.3.1 Preprocessing

Data pre-processing is an important step in the data mining process. The phrase "garbage in, garbage out" is particularly applicable to data mining and machine learning projects. Data-gathering methods are often loosely controlled, resulting in out-of-range values (e.g., Income: −100), impossible data combinations missing values, etc. Analyzing data that has not been carefully screened for such problems can produce misleading results. Thus, the representation and quality of data is first and foremost before running an analysis. Often, data pre-processing is the most important phase of a machine learning project, especially in computational biology. The product of data pre- processing is the final training set.

4.3.2 Feature Extraction

Feature extraction involves reducing the number of resources required to describe a large set of data. When performing analysis of complex data one of the major problems stems from the number of variables involved. Analysis with a large number of variables generally requires a large amount of memory and computation power, also it may cause a classification algorithm to over fit to training samples and generalize poorly to new samples. Feature extraction is a general term for methods of constructing combinations of the variables to get around these problems while still describing the data with sufficient accuracy. Many machine learning practitioners believe that properly optimized feature extraction is the key to effective model construction.

4.3.3 Feature Selection

Feature selection, also known as variable selection, attribute selection or variable subset selection, is the process of selecting a subset of relevant features (variables, predictors) for use in model construction. Feature selection techniques are used for several reasons:

Simplification of models to make them easier to interpret by researchers/users.

Shorter training times.

To avoid the curse of dimensionality.

Enhanced generalization by reducing over fittsing by reducing over fitting.

4.3.4 Prediction

Predictive analytics is driven by predictive modelling. It's more of an approach than a process. Predictive analytics and machine learning go hand-in-hand, as predictive models typically include a machine learning algorithm. These models can be trained over time to respond to new data or values, delivering the results the business needs. Predictive modelling largely overlaps with the field of machine learning.

Patterned after the operation of neurons in the human brain, neural networks (also called artificial neural networks) are a variety of deep learning technologies. They're typically used to solve complex pattern recognition problems – and are incredibly useful for analysing large data sets. They are great at handling nonlinear relationships in data – and work well when certain variables are unknown. This is the last step of our project. Here the we can identify the image weather it is medicinal or non-medicinal.

CHAPTER 5

RESULTS AND DISCUSSIONS

5.1 Front End Module

5.1.1 Home Module

The home page serves as the entry point of the web-based application for medicinal plant classification. It offers a visually appealing and informative overview of the system's objective identifying medicinal plants using machine learning techniques. A brief description highlights the use of Convolutional Neural Networks (CNNs) for accurate classification based on leaf image analysis. The interface includes key navigation elements such as links to the login page and an abstract button that introduces users to the core concept of the project. Additional design elements, like the background imagery and styled headings, enhance user engagement. This page sets the stage for a seamless experience in exploring the AI-based plant identification system.

Fig 5.1: Home Page

5.1.2 Abstract Module

The abstract module provides a concise yet impactful summary of the project's core objective. It introduces users to the significance of medicinal plant identification and the challenges involved in traditional methods. The content highlights how the integration of machine learning, particularly classifiers like ANN, CNN, and KNN, can overcome issues such as misidentification and time-consuming manual processes. It also emphasizes the role of image acquisition tools in generating structured datasets, which improve the accuracy and efficiency of classification. Displayed over a thematic background, this module adds educational value while preparing users for the technical depth explored in subsequent sections. The “Abstract” section is accessible directly from the homepage, ensuring easy navigation.

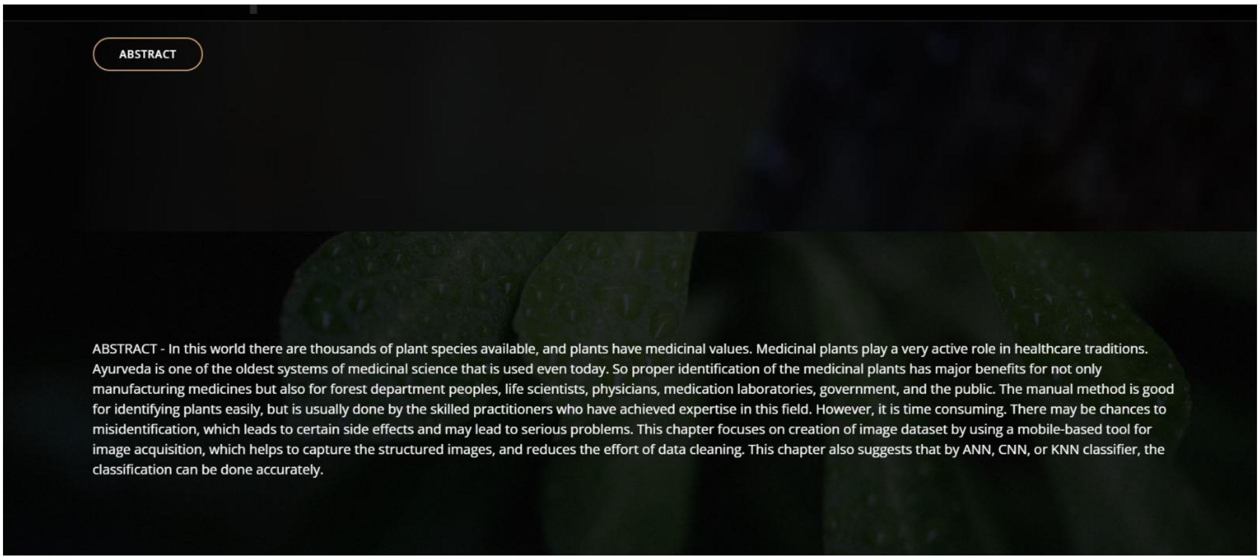


Fig 5.2: Introduction about Medicinal Plants

5.1.3 Login Module

The login module serves as a secure entry point for authorized users of the medicinal plant classification system. It presents a simple and intuitive interface, prompting users to enter their username and password for authentication. The clean layout ensures a smooth user experience while maintaining the professional look of the application. This module is essential for protecting access to sensitive functionalities such as image uploads, training results, and dataset management. By restricting access to verified users, the system ensures data integrity and controlled usage. The login button redirects authenticated users to the core features of the application, supporting a seamless navigation flow.

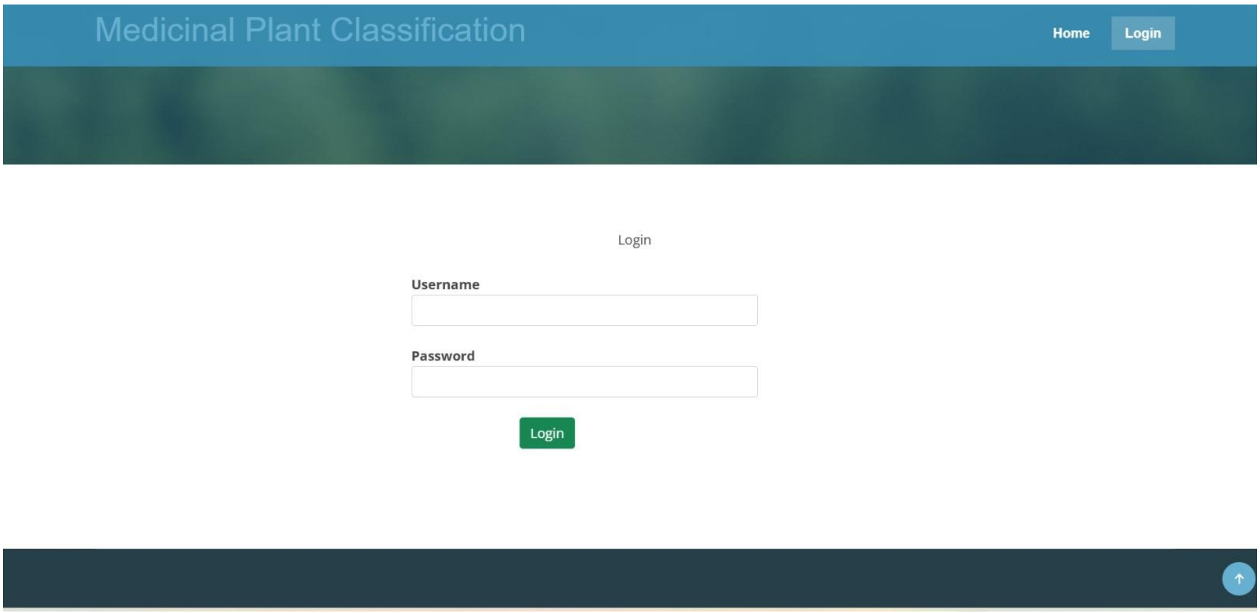


Fig 5.3: Login Interface

5.2 Image Uploading Module

The image uploadation module is a central feature of the medicinal plant classification system, enabling users to input leaf images for identification. The interface prominently displays a button labeled “Choose Plant Photo,” which allows users to select images directly from their local device. This module supports structured and simplified data acquisition, ensuring that the classification process begins with user-supplied input. Upon uploading, the system processes the image using deep learning models trained on plant datasets. The design maintains a clean aesthetic, helping users focus on the task with minimal distraction. This module acts as the first step in engaging with the classifier, seamlessly integrating user interaction with AI functionality.

Fig 5.4: Image Uploading Interface

5.2.1 Result Module

Tulsi has also been shown to counter metabolic stress through normalization of blood glucose, blood pressure and lipid levels, and psychological stress through positive effects on memory and cognitive function and through its anxiolytic and anti-depressant properties.

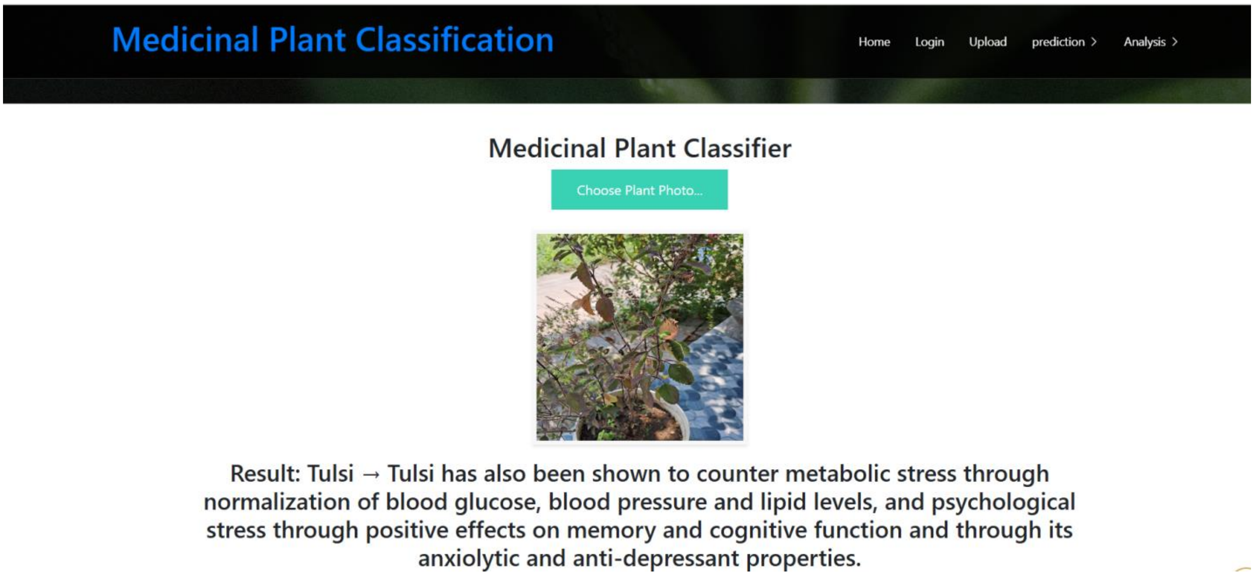


Fig 5.5: Plant name and Its uses Interface

5.2.2 Bar Chart Module

Among the medicinal plants analyzed, Abram is the most potent, curing 102 diseases, far exceeding the others. Indian Wormwood (28 diseases), Avoceva (22), and Nirgundi (18) follow at a considerable distance. The remaining plants—Mexican Mint**,** Kokilaksha**,** Neem, and Tulsi—show moderate to lower effectiveness, curing between 12 and 17 diseases each.

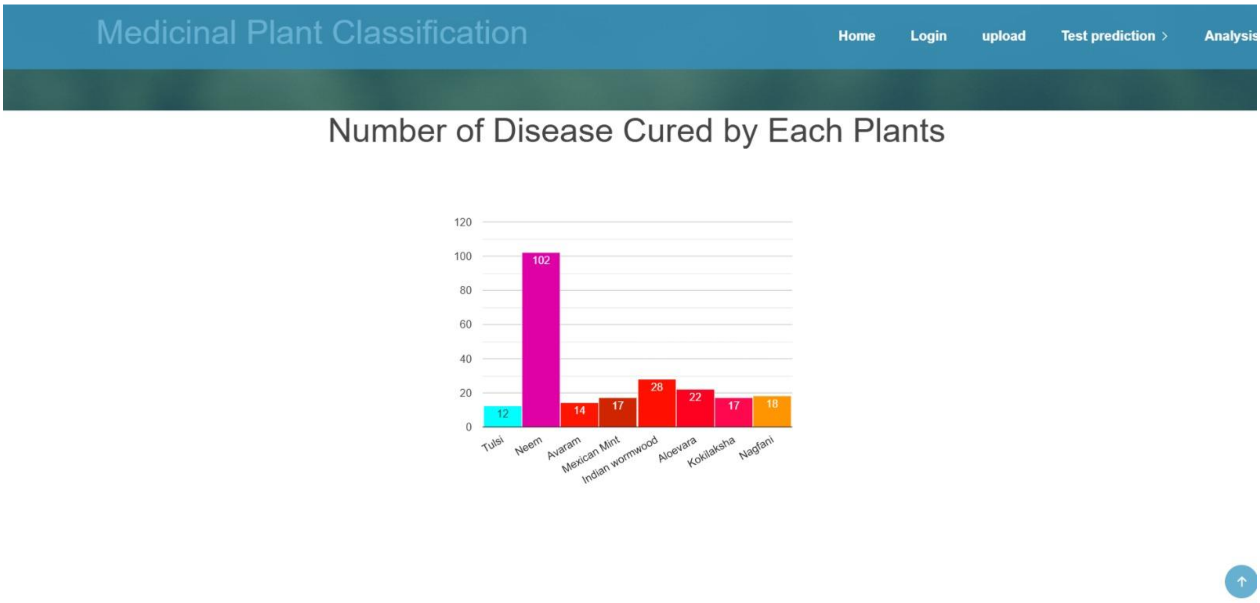


Fig 5.6: Disease Analysis Interface

The obtained results of training accuracy and validation accuracy values are shown in following graph.

In this diagram for where there has less number of epochs (upto 13) we can know that validation accuracy is same as the training accuracy. In case if we need more number of epochs the training accuracy and validation accuracy values are almost comparable.

5.2.3 Accuracy Module

The model shows high accuracy for both training and validation data over 50 epochs.

- Accuracy quickly rises above 90% within the first few epochs.
- It stabilizes near 100%, with minimal difference between training and validation curves.
- This indicates that the model is well-trained with minimal overfitting and strong generalization performance.

Accuracy Plot

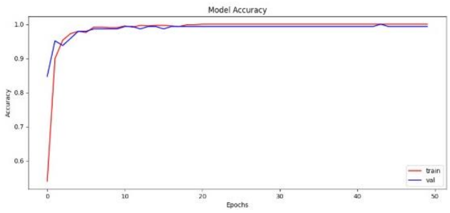


Fig 5.7 : Training accuracy vs validation accuracy graph

5.3 USES OF MEDICINAL PLANTS

This plant database is a curated collection of various species used for medicinal plant identification, particularly in traditional and herbal medicine. It includes a wide range of botanical names such as Aloe vera, Holy Basil (Tulsi), Neem, and Rosary Pea, alongside lesser-known regional species like Avaram, Nagfani, and Nalta Jute. Each plant entry may be associated with images, features, and local names for effective visual recognition and classification. The database is likely structured to support Convolutional Neural Network **(CNN)** models by providing labeled plant images. It encompasses plants with diverse medicinal uses including anti-inflammatory, antibacterial, and antipyretic properties. With references to local and regional classifications, the database can be valuable in community health research. It appears to integrate file directories and image repositories for training and testing machine learning models. Additionally, it supports leaf-based feature extraction**,** which is critical for species with similar structures. The inclusion of both common and scientific names enhances usability across disciplines. Overall, this database is an essential asset.

CHAPTER 6
CONCLUSION

In this work, we addressed the problem of identifying the medicinal plant species by the analysis of leaf images obtained directly from their habitat and irrespective of lighting conditions. The fixed zero threshold, vegetative index is successfully tested for image dataset. The result shows that the algorithm can adequately segment the leaf region. This method worked well in images with reflection. The feature extraction based on the color and texture features is done. The classification of medicinal plant species is done by using CNN and optained accuracy of 98.3% is measured. In future we have planned to design and develop a system which automatically identifies plant species through the analysis of not only the leaf images also the other parts of the plant acquired directly in their habitat irrespective of complex backgrounds and various lighting condition. This excellent performance indicates the viability of such computer-aided approaches in the classification of biological specimens and its potential applicability in combatting the 'taxonomic crists'. A web-based or mobile computer system for 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 also contribute significantly in the protection of endangered species. It is necessary to study about medicinal plants that find place in folklore. As a result, such plants should be investigated to understand their physical and chemical properties as well as safety and efficacy. The medicinal plants find applications in many industries. The use of medicinal plants for curing diseases has been increased. In order to identify the medicinal or non- medicinal it will take more time. So that we can create an application by using this project. Then, it will be easy to identify the medicinal plants. It is easy to use and also takes less time.

REFERENCES

[1] Amin, A.H.M. and Khan, A.I., 2013. One- Shot Classification of 2-D Leaf Shapes using Distributed Hierarchical Graph Neuron (DHGN) Scheme with a k-NN Classifier. Procedia Computer Science, 24, 84-96.

[2] Arai, K., Abdullah, I.N. and Okumura, H., 2013. Identification of Ornamental Plant Functioned as Medicinal Plant-Based on Redundant Discrete Wavelet Transformation. International Journal of Advanced Research in Artificial Intelligence, 2(3), 60-64.

[3] Babatunde, A., Armstrong, L., Diepeveen,D. and Leng, J., 2015. A survey of computer-based vision systems for automatic identification of plant species. Journal of Agricultural Informatics, 6(1), 61-71.

[4] Backes, A.R., Casanova, D. and Bruno, O.M., 2009. Plant Leaf Identification based on Volumetric Fractal Dimension. International Journal of Pattern Recognition and Artificial Intelligence, 23(6), 145- 1160.

[5] Carranza-Rojas, J. and Mata-Montero, E., 2016. Combining Leaf Shape and Texture for Costa Rican Plant Species Identification. CLEI Electronic Journal, 19(1), Paper 7.

[6] Chaki, J., Parekh, R. and Bhattacharya, S, 2015. Plant leaf recognition using texture and shape features with neural classifiers. Pattern Recognition Letters, 58, 61-68.

[7] Du, J.X., Wang, X.F. and Zhang, G.J., 2007. Leaf shape based plant species recognition. Applied Mathematics and Computation, 185, 883- 893.

[8] Du, J.X., Zhai, C.M. and Wang, Q.P., 2013. Recognition of plant leaf image based on fractal dimension features. Neurocomputing, 116, 150- 156.

[9] Du, M., Zhang, S. and Wang, H., 2009. Supervised Isomap for Plant Leaf Image Classification. 5th International Conference on Emerging Intelligent Computing Technology and Applications, Ulsan, South Korea, 627-634.

[10] Gao, W. and Lin, W., 2012. Frontal Parietal Control Network Regulates the Anti-Correlated Default and Dorsal Attention Networks. Human Brain Mapping, 33(1), 192–202.

[11] Herdiyeni, Y. and Wahyuni, N.K.S., 2012. Mobile Application for Indonesian Medicinal Plants Identification using Fuzzy Local Binary Pattern and Fuzzy Color Histogram. International Conference on Advanced Computer Science and Information Systems (ICACSIS), West Java, Indonesia, 301-306.

[12] Hernandez-Serna, A. and Jimenez-Segura, L.F., 2014. Automatic Identification of

species with neural networks. PeerJ 2:e563; DOI:10.7717/peerj.563.

[13] Hossain, J. and Amin, M.A., 2010. Leaf Shape Identification Based Plant Biometrics. 13th International Conference on Computer and Information Technology, Dhaka, Bangladesh, 458-463.

[14] Le, T.L., Tran, D.T. and Hoang, V.N., 2014. Fully Automatic leaf-based plant identification, application for Vietnamese medicinal plant search. Fifth Symposium on Information and Communication Technology, Hanoi, Vietnam, 146-154.

[15] Mata-Montero, E. and Carranza-Rojas, J., 2016. Automated Plant Species Identification: Challenges and Opportunities. IFIP Advances in Information and Communication Technology, 481, 26-36.

[16] Munisami, T., Ramsurn, M., Krishnan, S. and Pudaruth, S., 2015. Plant leaf recognition using shape features and color histogram with k- nearest neighbor classifiers. Procedia Computer Science, 58, 740-747.

[17] Prasvta, D.S. and Herdiyeni, Y., 2013. Red Leaf: Mobile Application for Medicinal Plant Identification Based on Leaf Image. International Journal of Advanced Science, Engineering and Information Technology, 3, 5– 8.

[18] Siravenha, A.C.Q. and Carvalho, S.R., 2015. Exploring the use of Leaf Shape Frequencies for Plant Classification. 28th SIBGRAPI Conference on Graphics, Patterns, and Images, Salvador, Brazil, 297-304.

[19] Wu, S.G., Bao, F.S., Xu, E.Y., Wang, Y.X., Chang, Y.F. and Xiang, Q.L., 2007. A Leaf Recognition Algorithm for Plant Classification using Probabilistic Neural Network. 7th IEEE International Symposium on Signal Processing and Information Technology, Giza, Egypt, 11- 16.

APPENDIX A

SOURCE CODE

Training The CNN Model

```
import numpy as np
import pandas as pd
import tensorflow
import matplotlib.pyplot as plt
from matplotlib.image import imread
import cv2
import os
from os import listdir
from PIL import Image
from sklearn.preprocessing import label_binarize,  LabelBinarizer
from tensorflow.keras.preprocessing import image
from tensorflow.keras.preprocessing.image import img_to_array,     array_to_img
from tensorflow.keras.optimizers import Adam
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Conv2D, MaxPooling2D
from tensorflow.keras.layers import Activation, Flatten, Dropout, Dense
from sklearn.model_selection import train_test_split
from tensorflow.keras.models import model_from_json
from tensorflow.keras.utils import to_categorical
import logger
from preprocessing import Preprocessor
import datapath as data

class trainModel:
    def __init__(self):
        self.log_writer = logger.App_Logger()
        self.file_object = open("../Training_Logs/ModelTrainingLog.txt", 'a+')
        self.Preprocessor = Preprocessor()
```

```
def trainingModel(self):
    # Logging the start of training
    self.log_writer.log(self.file_object, 'Start of Training')
    try:
        """doing the preprocessing"""

        self.log_writer.log(self.file_object, 'Doing Preprocessing.')
        # Doing necessary preprocessing
        self.log_writer.log(self.file_object, 'I enter Preprocessing.')



        dir = "../dataset"
        root_dir = listdir(dir)
        image_list, label_list = [], []
        all_labels = ['dyed-lifted-polyps', 'dyed-resection-margins', 'esophagitis']
        binary_labels = [0,1,2]
        temp = -1

        # Reading and converting image to numpy array
        for directory in root_dir:
            plant_image_list = listdir(f"{dir}/{directory}")
            temp += 1
            for files in plant_image_list:
                image_path = f"{dir}/{directory}/{files}"
                image_list.append(self.Preprocessor.convert_image_to_array(image_path))
                label_list.append(binary_labels[temp])

        # splitting the data into training and test set
        self.log_writer.log(self.file_object, 'Doing train_test_split.')
        x_train, x_test, y_train, y_test = train_test_split(image_list, label_list, test_size=0.2,
```

```
random_state = 10)
        self.log_writer.log(self.file_object, 'Finish train_test_split.')

        # Normalizing the dataset
        self.log_writer.log(self.file_object, 'Doing Normalizing.')
        x_train = np.array(x_train, dtype=np.float16) / 225.0
        x_test = np.array(x_test, dtype=np.float16) / 225.0
        x_train = x_train.reshape( -1, 256,256,3)
        x_test = x_test.reshape( -1, 256,256,3)

        # Changing to categorical
        self.log_writer.log(self.file_object, 'Changing to categorical.')
        y_train = to_categorical(y_train)
        y_test = to_categorical(y_test)

        # Model Building
        self.log_writer.log(self.file_object, 'Model Building.')
        model = Sequential()
        model.add(Conv2D(32, (3, 3), padding="same",input_shape=(256,256,3),
activation="relu"))
        model.add(MaxPooling2D(pool_size=(3, 3)))
        model.add(Conv2D(16, (3, 3), padding="same", activation="relu"))
        model.add(MaxPooling2D(pool_size=(2, 2)))
        model.add(Flatten())
        model.add(Dense(8, activation="relu"))
        model.add(Dense(3, activation="softmax"))
        model.summary()



        # Compiling Modell
        self.log_writer.log(self.file_object, 'Compiling Model.')
```

```
model.compile(loss = 'categorical_crossentropy', optimizer = Adam(0.0001),
metrics=['accuracy'])

# Again splitting the training dataset into training and validation datasets
self.log_writer.log(self.file_object, 'Again doing train_test_split.')
x_tr, x_val, y_tr, y_val = train_test_split(x_train, y_train, test_size = 0.2)

# Training the model
epochs = 20
batch_size = 20
self.log_writer.log(self.file_object, 'Fitting a model.')
history = model.fit(x_tr, y_tr, batch_size = batch_size, epochs = epochs, validation_data
= (x_val, y_val), shuffle=True)

# Saving Model
self.log_writer.log(self.file_object, 'Saving Model.')
model.save(r"C:\Users\way2m\OneDrive\Documents\Python Projects\IBS Project
COde\IBS_RCNN\dataset\disease.h5")
# Serialize model to json
json_model = model.to_json()
# save the model architechture to JSON file
with open("../dataset/disease.json", "w") as json_file:
   json_file.write(json_model)

#saving the weights of the model
model.save_weights(r"C:\Users\way2m\OneDrive\Documents\Python Projects\IBS
Project COde\IBS_RCNN\dataset\disease_weights.h5")

#Plot the training history
self.log_writer.log(self.file_object, 'Plotting the training history.')
plt.figure(figsize=(12, 5))
```

```
            plt.plot(history.history['accuracy'], color='r')
            plt.plot(history.history['val_accuracy'], color='b')
            plt.title('Model Accuracy')
            plt.ylabel('Accuracy')
            plt.xlabel('Epochs')
            plt.legend(['train', 'val'])
            plt.show()
            plt.savefig("../plot/plot.png")   # save the figure to file

            # logging the successful Training
            self.log_writer.log(self.file_object, 'Successful End of Training')
            self.file_object.close()

        except Exception as e:
            # logging the unsuccessful Training
            self.log_writer.log(self.file_object, "Exception occured in tainingModel of trainModel
class. Exception message:  "+str(e))
            self.log_writer.log(self.file_object, "trainingModel Unsuccessful. Exited the
trainingModel method of the trainModel class")
            raise Exception()
```

Main Code

```
from flask import *
import os
from werkzeug.utils import secure_filename
import label_image

def load_image(image):
    text = label_image.main(image)
    return text

app = Flask(__name__)

@app.route('/')
@app.route('/first')
def first():
    return render_template('first.html')



@app.route('/login')
def login():
    return render_template('login.html')

@app.route('/chart')
def chart():
    return render_template('chart.html')




@app.route('/index')
def index():
    return render_template('index.html')
```

```
@app.route('/predict', methods=['GET', 'POST'])
def upload():
    if request.method == 'POST':
        # Get the file from post request
        f = request.files['file']
        file_path = secure_filename(f.filename)
        f.save(file_path)
        # Make prediction
        result = load_image(file_path)
        result = result.title()
        d = {"Asthma Plant":" → Euphorbia hirta is a pantropical weed, originating from the
tropical regions of the Americas. It is a hairy herb that grows in open grasslands, roadsides
and pathways. It is widely used in traditional herbal medicine across many cultures,
particularly for asthma, skin ailments, and hypertension.",
            'Avaram':" → Senna auriculata is a leguminous tree in the subfamily
Caesalpinioideae. It is commonly known by its local names matura tea tree, avaram or
ranawara, or the English version avaram senna. It is the State flower of Telangana. It occurs
in the dry regions of India and Sri Lanka. ",
        "Balloon Vine":" → Cardiospermum halicacabum, known as the lesser balloon vine,
balloon plant or love in a puff, is a climbing plant widely distributed across tropical and
subtropical areas of Africa, Australia, and North America that is often found as a weed along
roads and rivers.",
        "Bellyache Bush":" → Jatropha gossypiifolia, commonly known as bellyache bush,
black physicnut or cotton-leaf physicnut, is a species of flowering plant in the spurge family,
Euphorbiaceae. The species is native to Mexico, Philippines, South America, Gujarat State,
and the Caribbean islands. ",
        "Benghal Dayflower":" → Commelina benghalensis, commonly known as the Benghal
dayflower, tropical spiderwort, or wandering Jew, kanshira in Bengali, is a perennial herb
native to tropical Asia and Africa",
        "Big Caltrops":" → A caltrop also known as caltrap, galtrop, cheval trap, galthrap,
galtrap, calthrop, jackrock or crow's foot is an area denial weapon made up of two or more
sharp nails or spines arranged in such a manner that one of them always points upward from a
stable base for example, a tetrahedron. Historically, caltrops were part of defences that served
to slow the advance of troops, especially horses, chariots, and war elephants, and were
```

particularly effective against the soft feet of camels. In modern times, caltrops are effective when used against wheeled vehicles with pneumatic tires.",

"Black Honey Shrub":" → Black-Honey Shrub is usually a much-branched somewhat climbing shrub, rarely a small tree. Leaves are ovate-oblong to elliptic, 1-5 cm long, 0.7-3 cm wide, produced on short lateral branchlets, looking like leaflets of a compound leaf.",

"Bristly Wild Grape":" → Cyphostemma setosum - Bristly Wild Grape. Bristly Wild Grape is a succulent climber clothed with bristly hairs. Leaves are trifoliate rarely lower ones simple, fleshy, nearly stalkless, shortly stalked, elliptic, ovoate to obovate, blunt, rounded at the base, stringly sawtoothed, up to 7 x 5 cms.31-Dec-2016",

"Butterfly Pea":" → Clitoria ternatea, commonly known as Asian pigeonwings, bluebellvine, blue pea, butterfly pea, cordofan pea or Darwin pea is a plant species belonging to the family Fabaceae, endemic and native to the Indonesian island of Ternate. In India, it is revered as a holy flower, used in daily puja rituals.",

'Cape Gooseberry':" → Physalis peruviana, is a South American plant native to Colombia, Ecuador and Peru in the nightshade family, commonly known as Cape gooseberry or goldenberry, known in its countries of origin as ",

"Common Wireweed":" → Sida acuta, the common wireweed, is a species of flowering plant in the mallow family, Malvaceae. It is believed to have originated in Central America, but today has a pantropical distribution and is considered a weed in some areas.",

"Country Mallow":" → Sida cordifolia is a perennial subshrub of the mallow family Malvaceae native to India. It has naturalized throughout the world, and is considered an invasive weed in Africa, Australia, the southern United States, Hawaiian Islands, New Guinea, and French Polynesia.",

"Crown Flower":" → Calotropis gigantea, the crown flower, is a species of Calotropis native to Cambodia, Vietnam, Bangladesh, Indonesia, Malaysia, Thailand, Sri Lanka, India, China, Pakistan, and Nepal. It is a large shrub growing to 4 m tall. It has clusters of waxy flowers that are either white or lavender in colour.",

"Green Chireta":" → Andrographis paniculata, commonly known as creat or green chiretta, is an annual herbaceous plant in the family Acanthaceae, native to India and Sri Lanka. It is widely cultivated in Southern and Southeastern Asia, where it has been believed to be a treatment for bacterial infections and some diseases.",

"Holy Basil":" → Ocimum tenuiflorum, commonly known as holy basil, tulsi or tulasi, is an aromatic perennial plant in the family Lamiaceae. It is native to the Indian subcontinent and widespread as a cultivated plant throughout the Southeast Asian tropics", "Indian Copperleaf":" → Acalypha indica is an herbaceous annual that has catkin-like inflorescences with cup-shaped involucres surrounding the minute flowers. It is mainly known for its root being attractive to domestic cats, and for its various medicinal uses. It occurs throughout the Tropics. ",

"Indian Jujube":" → Ziziphus mauritiana, also known as Indian jujube, Indian plum, Chinese date, Chinese apple, ber, and dunks is a tropical fruit tree species belonging to the family Rhamnaceae. ",

'Indian Sarsaparilla':" → Hemidesmus indicus, Indian sarsaparilla is a species of plant found in South Asia. It occurs over the greater part of India, from the upper Gangetic plain eastwards to Assam and in some places in central, western and South India. The root is a substitute for sarsaparilla.",

"Indian Stinging Nettle":" → Tragia involucrata, the Indian stinging nettle, is a species of plant in the family Euphorbiaceae.",

"Indian Thornapple":" → Datura innoxia, known as pricklyburr, recurved thorn-apple, downy thorn-apple, Indian-apple, lovache, moonflower, nacazcul, toloatzin, toloaxihuitl, tolguache or toloache, is a species of flowering plant in the family Solanaceae. ",

"Indian Wormwood":" → Indian Wormwood is found in India and East Asia. It is found in the Himalayas at altitudes of 300-2400 m. Flowering: August-October. Medicinal uses: An infusion of leaves is used in the treatment of nervous and spasmodic affections, in asthma and in diseases of the brain.",

"Ivy Gourd":" → Ivy gourd is a plant. The leaves, root, and fruit are used to make medicine. Ivy gourd is most often used for diabetes. People also use ivy gourd for gonorrhea, constipation, wounds, and other conditions, but there is no good scientific evidence to support these uses.",

"Kokilaksha":" → The plant is an Ayurvedic herb used to make medicines for several gastrointestinal, kidney, reproductive, liver, and bone disorders. Kokilaksha is native to India and also to other places like Srilanka, Malaysia, Nepal and Myanmar. Kokilaksha means having eyes like Kokila the Indian cuckoo.",

"Land Caltrops":" → Tribulus terrestris is an annual plant in the caltrop family widely distributed around the world. It is adapted to thrive in dry climate locations in which few other plants can survive. It is native to warm temperate and tropical regions in southern Eurasia and Africa. ",

"Madagascar Periwinkle":" → Catharanthus roseus, commonly known as bright eyes, Cape periwinkle, graveyard plant, Madagascar periwinkle, old maid, pink periwinkle, rose periwinkle, is a species of flowering plant in the family Apocynaceae. It is native and endemic to Madagascar, but grown elsewhere as an ornamental and medicinal plant. ",

'Madras Pea Pumpkin':" → Madras pea pumpkin is a perennial herb, climbing or trailing up to 3 m, stem bristly-hairy. Tendrils are simple, thread-like. Leaves are arrow shaped, hastate, ...",

"Malabar Catmint":" → Anisomeles indica, or catmint, is a species of herbaceous plant native to eastern Asia and naturalized on some Pacific islands.",

"Mexican Mint":" → Coleus amboinicus, synonym Plectranthus amboinicus, is a semi-succulent perennial plant in the family Lamiaceae with a pungent oregano-like flavor and odor",

"Mexican Prickly Poppy":" → Argemone mexicana is a species of poppy found in Mexico and now widely naturalized in many parts of the world. An extremely hardy pioneer plant, it is tolerant of drought and poor soil, often being the only cover on new road cuttings or verges. It has bright yellow latex. ",

"Mountain Knotgrass":" → Aerva lanata, the mountain knotgrass, is a woody, prostrate or succulent, perennial herb in the family Amaranthaceae, native to Asia, Africa. It has been included as occurring in Australia by the US government, but it is not recognised as occurring in Australia by any Australian state herbarium.",

"Nalta Jute":" → Jute mallow or nalta jute is a species of shrub in the family Malvaceae. Together with C. capsularis it is the primary source of jute fiber. The leaves and young fruits are used as a vegetable, the dried leaves are used for tea and as a soup thickener, and the seeds are edible.",

"Night Blooming Cereus":" → Regardless of genus or species, night-blooming cereus flowers are almost always white or very pale shades of other colors, often large, and frequently fragrant. Most of the flowers open after nightfall, and by dawn, most are in the process of wilting.",

"Panicled Foldwing":" → Dicliptera paniculata - Panicled Foldwing. Panicled Foldwing is an erect herb, 0.6-1.2 m tall. Young shoots are usually 4-sided; adult shoots 6-sided, white spreading bristle-hairy. Ovate leaves opposite, equal and unequal; leaf-stalk 3-5 mm.",

"Prickly Chaff Flower":" → Achyranthes aspera is a species of plant in the family Amaranthaceae. It is distributed throughout the tropical world. It can be found in many places growing as an introduced species and a common weed. It is an invasive species in some areas, including many Pacific Islands environments.",

"Punarnava":" → Boerhaavia diffusa is a species of flowering plant in the four o'clock family which is commonly known as punarnava, red spiderling, spreading hogweed, or tarvine. It is taken in herbal medicine for pain relief and other uses. The leaves of Boerhaavia diffusa are often used as a green vegetable in many parts of India.",

'Purple Fruited Pea Eggplant':" → It is a very good brain stimulant. It removes chest, nose and head congestion. It relieves cough and cold. It relieves dyspepsia impairment of digestion",

"Purple Tephrosia":" → Tephrosia purpurea is a species of flowering plant in the family Fabaceae, that has a pantropical distribution. It is a common wasteland weed. In many parts it is under cultivation as green manure crop. It is found throughout India and Sri Lanka in poor soils. Common names include: Bengali",

"Rosary Pea":" → Abrus precatorius, commonly known as jequirity bean or rosary pea, is a herbaceous flowering plant in the bean family Fabaceae. It is a slender, perennial climber with long, pinnate-leafleted leaves that twines around trees, shrubs, and hedges. ",

"Shaggy Button Weed":" → Landrina, Spermacoce hispida, SHAGGY BUTTON WEED - Herbal Medicine - An illustrated compilation of Philippine medicinal plants by Dr Godofredo Umali Stuart .",

"Small Water Clover":" → Water clovers are fast growers, with vestita and quadrifolia ranging between 4 inches and one foot of maximum height, depending on water depth ...",

"Spiderwisp":" → It is known by many common names including Shona cabbage, African cabbage, spiderwisp, cat's whiskers, chinsaga and stinkweed. It is an annual wildflower native to Africa but has become widespread in many tropical and sub-tropical parts of the world.",

"Square Stalked Vine":" → This site makes an attempt to gather and share common names of the plants found in India. The common names are just as important as the scientific names.",

"Stinking Passionflower":" → Native to South America, stinking passionflower is a climbing vine with an unpleasant smell and flowers that resemble those of the passionfruit vine. Stinking passionflower can invade forest edges, coastal vegetation, roadsides and disturbed areas. It is widespread in northern Queensland.",

"Sweet Basil":" → Basil, also called great basil, is a culinary herb of the family Lamiaceae. It is a tender plant, and is used in cuisines worldwide. In Western cuisine, the generic term basil refers to the variety also known as sweet basil or Genovese basil. Basil is native to tropical regions from Central Africa to Southeast Asia.",

"Sweet Flag":" → Acorus calamus is a species of flowering plant with psychoactive chemicals. It is a tall wetland monocot of the family Acoraceae, in the genus Acorus",

"Tinnevelly Senna":" → Senna alexandrina is an ornamental plant in the genus Senna. It is used in herbalism. It grows natively in upper Egypt, especially in the Nubian region, and near Khartoum, where it is cultivated commercially. It is also grown elsewhere, notably in India and Somalia. ",

'Trellis Vine':" → Unlike a tree, vines can't support themselves, so a trellis provides this support. Also, trellises keep vines off the ground and therefore minimize disease. They help spread out the canopy for sun exposure, pruning and canopy management.",

"Velvet Bean":" → Mucuna pruriens is a tropical legume native to Africa and tropical Asia and widely naturalized and cultivated. Its English common names include monkey tamarind, velvet bean, Bengal velvet bean, Florida velvet bean, Mauritius velvet bean, Yokohama velvet bean, cowage, cowitch, lacuna bean, and Lyon bean. ",

"Aloevara":" → Aloe vera is a succulent plant species of the genus Aloe. It is widely distributed, and is considered an invasive species in many world regions. An evergreen perennial, it originates from the Arabian Peninsula, but grows wild in tropical, semi-tropical, and arid climates around the world",

"Coatbuttons":" → Tridax procumbens, commonly known as coatbuttons or tridax daisy, is a species of flowering plant in the family Asteraceae. It is best known as a widespread weed and pest plant.",

"Heart Leaved Moonseed":" → Tinospora cordifolia is a herbaceous vine of the family Menispermaceae indigenous to tropical regions of the Indian subcontinent. It has been used in Ayurveda to treat various disorders, but there is no clinical evidence for the effectiveness of such treatment.",

"Nagfani":" → Crataegus, commonly called hawthorn, quickthorn, thornapple, May-tree, whitethorn, Mayflower, or hawberry, is a genus of several hundred species of shrubs and trees in the family Rosaceae, native to temperate regions of the Northern Hemisphere in Europe, Asia, North Africa, and North America.",

"Neem":" → Azadirachta indica, commonly known as neem, nimtree or Indian lilac, is a tree in the mahogany family Meliaceae. It is one of two species in the genus Azadirachta, and is native to the Indian subcontinent. It is typically grown in tropical and semi-tropical regions. Neem trees also grow on islands in southern Iran.",

"Tulsi":" → Tulsi has also been shown to counter metabolic stress through normalization of blood glucose, blood pressure and lipid levels, and psychological stress through positive effects on memory and cognitive function and through its anxiolytic and anti-depressant

```
properties."}
        result = result+d[result]
        #result2 = result+d[result]
        #result = [result]
        #result3 = d[result]
        print(result)
        #print(result3)
        os.remove(file_path)
        return result
        #return result3
    return None
if __name__ == '__main__':
    app.run()
```

APPENDIX B

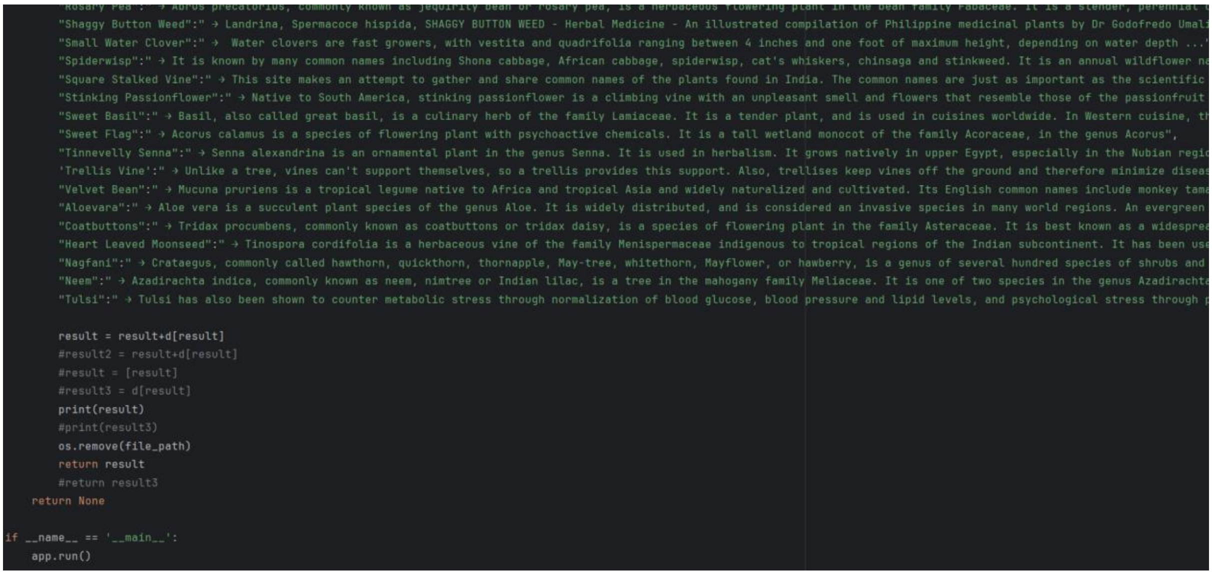
SCREENSHOTS

```
from flask import *
import os
from werkzeug.utils import secure_filename
import label_image

def load_image(image):
    text = label_image.main(image)
    return text

app = Flask(__name__)

@app.route('/')
@app.route('/first')
def first():
    return render_template('first.html')



@app.route('/login')
def login():
    return render_template('login.html')
@app.route('/chart')
def chart():
    return render_template('chart.html')


@app.route('/index')
def index():
    return render_template('index.html')


@app.route('/predict', methods=['GET', 'POST'])
```

Fig B1.1: Main Code

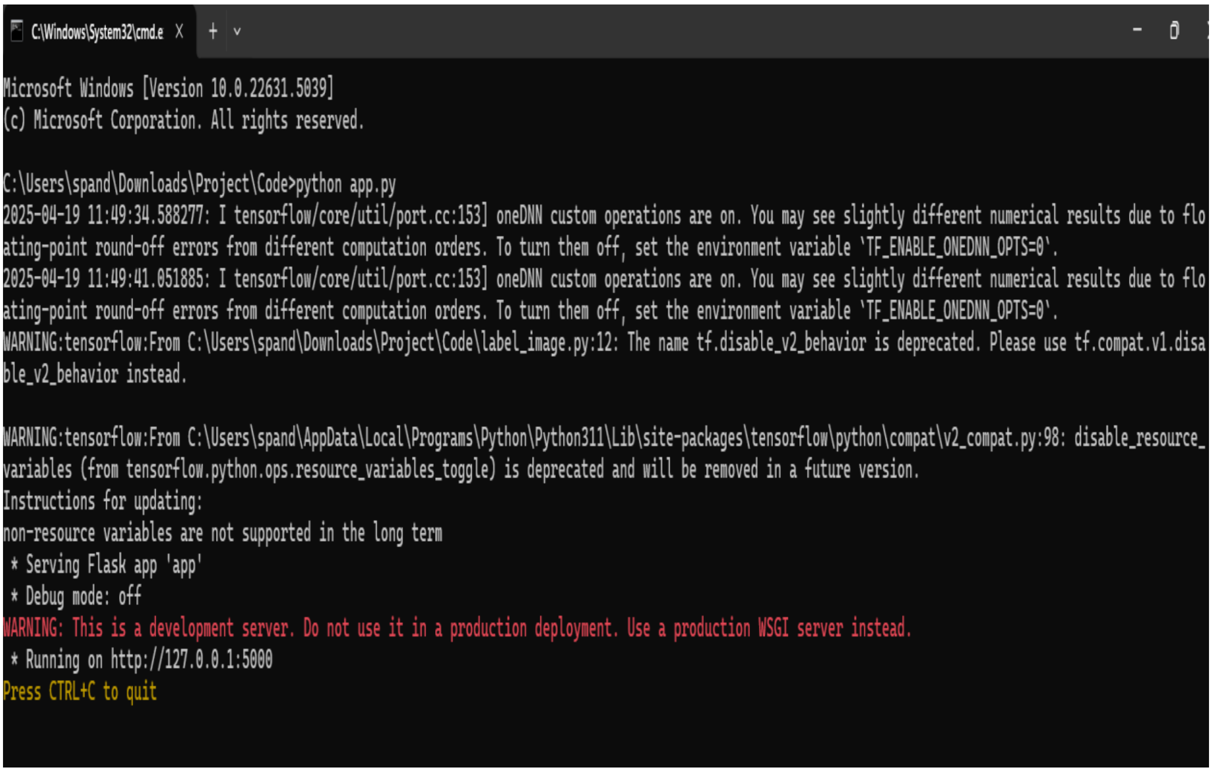
```
@app.route('/predict', methods=['GET', 'POST'])
def upload():
    if request.method == 'POST':
        # Get the file from post request
        f = request.files['file']
        file_path = secure_filename(f.filename)
        f.save(file_path)
        # Make prediction
        result = load_image(file_path)
        result = result.title()
        d = {"Asthma Plant":" → Euphorbia hirta is a pantropical weed, originating from the tropical regions of the Americas. It is a hairy herb that grows in open grasslands, road
    'Avaram':" → Senna auriculata is a leguminous tree in the subfamily Caesalpinioideae. It is commonly known by its local names matura tea tree, avaram or ranawara, or the Englis
        "Balloon Vine":" → Cardiospermum halicacabum, known as the lesser balloon vine, balloon plant or love in a puff, is a climbing plant widely distributed across tropical and
        "Bellyache Bush":" → Jatropha gossypiifolia, commonly known as bellyache bush, black physicnut or cotton-leaf physicnut, is a species of flowering plant in the spurge famil
        "Benghal Dayflower":" → Commelina benghalensis, commonly known as the Benghal dayflower, tropical spiderwort, or wandering Jew, kanshira in Bengali, is a perennial herb nat
        "Big Caltrops":" → A caltrop also known as caltrap, galtrop, cheval trap, galthrap, galtrap, calthrop, jackrock or crow's foot is an area denial weapon made up of two or mo
        "Black Honey Shrub":" → Black-Honey Shrub is usually a much-branched somewhat climbing shrub, rarely a small tree. Leaves are ovate-oblong to elliptic, 1-5 cm long, 0.7-3 c
        "Bristly Wild Grape":" → Cyphostemma setosum - Bristly Wild Grape. Bristly Wild Grape is a succulent climber clothed with bristly hairs. Leaves are trifoliate rarely lower
        "Butterfly Pea":" → Clitoria ternatea, commonly known as Asian pigeonwings, bluebellvine, blue pea, butterfly pea, cordofan pea or Darwin pea is a plant species belonging t
        'Cape Gooseberry':" → Physalis peruviana, is a South American plant native to Colombia, Ecuador and Peru in the nightshade family, commonly known as Cape gooseberry or gold
        "Common Wireweed":" → Sida acuta, the common wireweed, is a species of flowering plant in the mallow family, Malvaceae. It is believed to have originated in Central America
        "Country Mallow":" → Sida cordifolia is a perennial subshrub of the mallow family Malvaceae native to India. It has naturalized throughout the world, and is considered an i
        "Crown Flower":" → Calotropis gigantea, the crown flower, is a species of Calotropis native to Cambodia, Vietnam, Bangladesh, Indonesia, Malaysia, Thailand, Sri Lanka, Indi
        "Green Chireta":" → Andrographis paniculata, commonly known as creat or green chiretta, is an annual herbaceous plant in the family Acanthaceae, native to India and Sri Lan
        "Holy Basil":" → Ocimum tenuiflorum, commonly known as holy basil, tulsi or tulasi, is an aromatic perennial plant in the family Lamiaceae. It is native to the Indian subco
        "Indian Copperleaf":" → Acalypha indica is an herbaceous annual that has catkin-like inflorescences with cup-shaped involucres surrounding the minute flowers. It is mainly
        "Indian Jujube":" → Ziziphus mauritiana, also known as Indian jujube, Indian plum, Chinese date, Chinese apple, ber, and dunks is a tropical fruit tree species belonging to
    'Indian Sarsaparilla':" → Hemidesmus indicus, Indian sarsaparilla is a species of plant found in South Asia. It occurs over the greater part of India, from the upper Gangetic p
        "Indian Stinging Nettle":" → Tragia involucrata, the Indian stinging nettle, is a species of plant in the family Euphorbiaceae.",
        "Indian Thornapple":" → Datura innoxia, known as pricklyburr, recurved thorn-apple, downy thorn-apple, Indian-apple, lovache, moonflower, nacazcul, toloatzin, toloaxihuitl,
        "Indian Wormwood":" → Indian Wormwood is found in India and East Asia. It is found in the Himalayas at altitudes of 300-2400 m. Flowering: August-October. Medicinal uses: A
        "Ivy Gourd":" → Ivy gourd is a plant. The leaves, root, and fruit are used to make medicine. Ivy gourd is most often used for diabetes. People also use ivy gourd for gonorr
```

Fig B1.2: Main Code



```
        "Rosary Pea":" → Abrus precatorius, commonly known as jequirity bean or rosary pea, is a herbaceous flowering plant in the bean family Fabaceae. It is a slender, perennial
        "Shaggy Button Weed":" → Landrina, Spermacoce hispida, SHAGGY BUTTON WEED - Herbal Medicine - An illustrated compilation of Philippine medicinal plants by Dr Godofredo Umali
        "Small Water Clover":" →  Water clovers are fast growers, with vestita and quadrifolia ranging between 4 inches and one foot of maximum height, depending on water depth ..."
        "Spiderwisp":" → It is known by many common names including Shona cabbage, African cabbage, spiderwisp, cat's whiskers, chinsaga and stinkweed. It is an annual wildflower na
        "Square Stalked Vine":" → This site makes an attempt to gather and share common names of the plants found in India. The common names are just as important as the scientific
        "Stinking Passionflower":" → Native to South America, stinking passionflower is a climbing vine with an unpleasant smell and flowers that resemble those of the passionfruit
        "Sweet Basil":" → Basil, also called great basil, is a culinary herb of the family Lamiaceae. It is a tender plant, and is used in cuisines worldwide. In Western cuisine, th
        "Sweet Flag":" → Acorus calamus is a species of flowering plant with psychoactive chemicals. It is a tall wetland monocot of the family Acoraceae, in the genus Acorus",
        "Tinnevelly Senna":" → Senna alexandrina is an ornamental plant in the genus Senna. It is used in herbalism. It grows natively in upper Egypt, especially in the Nubian regio
        'Trellis Vine':" → Unlike a tree, vines can't support themselves, so a trellis provides this support. Also, trellises keep vines off the ground and therefore minimize diseas
        "Velvet Bean":" → Mucuna pruriens is a tropical legume native to Africa and tropical Asia and widely naturalized and cultivated. Its English common names include monkey tama
        "Aloevara":" → Aloe vera is a succulent plant species of the genus Aloe. It is widely distributed, and is considered an invasive species in many world regions. An evergreen
        "Coatbuttons":" → Tridax procumbens, commonly known as coatbuttons or tridax daisy, is a species of flowering plant in the family Asteraceae. It is best known as a widesprea
        "Heart Leaved Moonseed":" → Tinospora cordifolia is a herbaceous vine of the family Menispermaceae indigenous to tropical regions of the Indian subcontinent. It has been use
        "Nagfani":" → Crataegus, commonly called hawthorn, quickthorn, thornapple, May-tree, whitethorn, Mayflower, or hawberry, is a genus of several hundred species of shrubs and
        "Neem":" → Azadirachta indica, commonly known as neem, nimtree or Indian lilac, is a tree in the mahogany family Meliaceae. It is one of two species in the genus Azadiracht
        "Tulsi":" → Tulsi has also been shown to counter metabolic stress through normalization of blood glucose, blood pressure and lipid levels, and psychological stress through p

        result = result+d[result]
        #result2 = result+d[result]
        #result = [result]
        #result3 = d[result]
        print(result)
        #print(result3)
        os.remove(file_path)
        return result
        #return result3
    return None

if __name__ == '__main__':
    app.run()
```

Fig B1.3: Main Code



```
C:\Windows\System32\cmd.e  X  +  v

Microsoft Windows [Version 10.0.22631.5039]
(c) Microsoft Corporation. All rights reserved.

C:\Users\spand\Downloads\Project\Code>python app.py
2025-04-19 11:49:34.588277: I tensorflow/core/util/port.cc:153] oneDNN custom operations are on. You may see slightly different numerical results due to flo
ating-point round-off errors from different computation orders. To turn them off, set the environment variable `TF_ENABLE_ONEDNN_OPTS=0`.
2025-04-19 11:49:41.051885: I tensorflow/core/util/port.cc:153] oneDNN custom operations are on. You may see slightly different numerical results due to flo
ating-point round-off errors from different computation orders. To turn them off, set the environment variable `TF_ENABLE_ONEDNN_OPTS=0`.
WARNING:tensorflow:From C:\Users\spand\Downloads\Project\Code\label_image.py:12: The name tf.disable_v2_behavior is deprecated. Please use tf.compat.v1.disa
ble_v2_behavior instead.

WARNING:tensorflow:From C:\Users\spand\AppData\Local\Programs\Python\Python311\Lib\site-packages\tensorflow\python\compat\v2_compat.py:98: disable_resource_
variables (from tensorflow.python.ops.resource_variables_toggle) is deprecated and will be removed in a future version.
Instructions for updating:
non-resource variables are not supported in the long term
 * Serving Flask app 'app'
 * Debug mode: off
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
 * Running on http://127.0.0.1:5000
Press CTRL+C to quit
```

Fig B1.4:Unit Testing

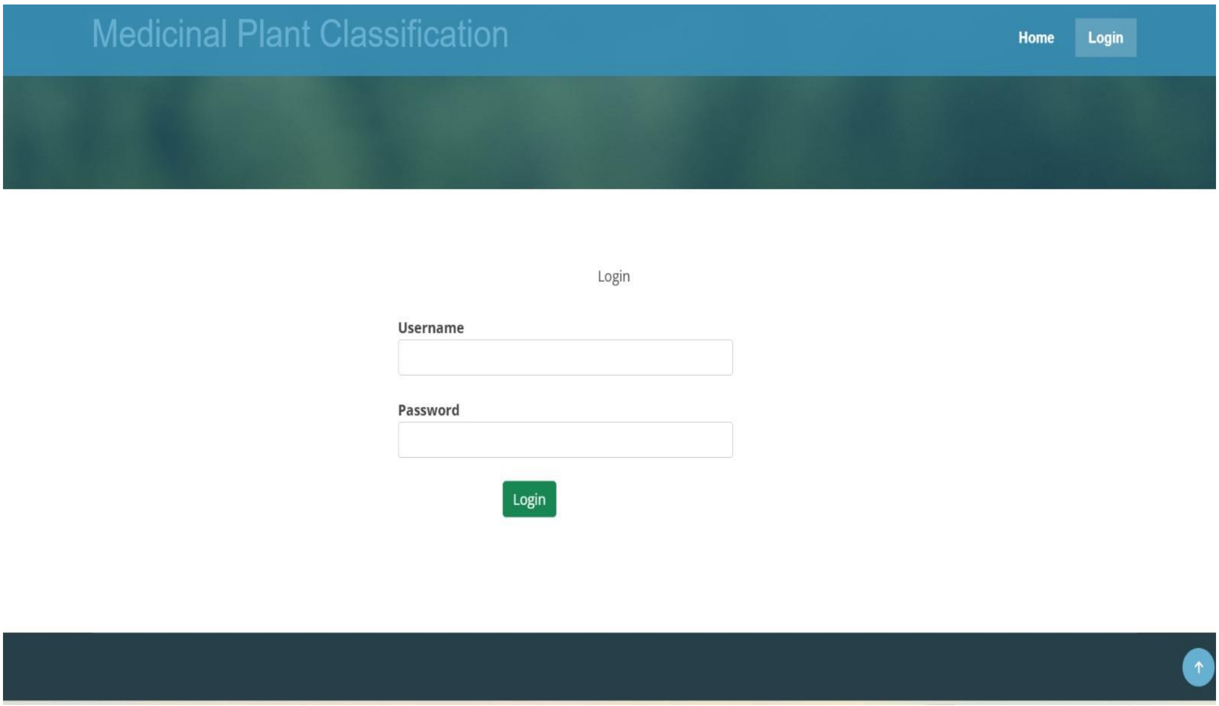


Fig B1.5: Login Interface

Fig B1.6: Uploading Image To Interface

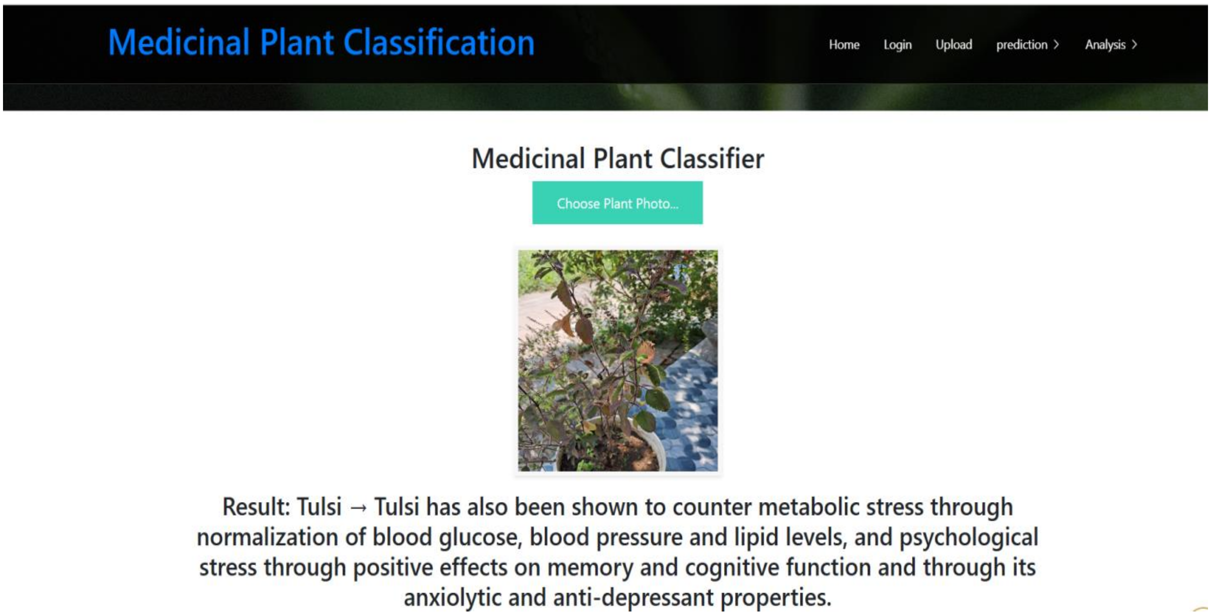


Fig B1.7: Result Page

APPENDIX C

STUDENT CONTRIBUTION

S.No.	ACTIVITY	21K61A6142	21K61A6129	21K61A6126	21K61A6136
1	Title Confirmation	✓	✓	✓	✓
2	Literature Survey	✓	✓	✓	✓
3	Problem Formulation	✓	✓	✓	✓
4	Requirement Gathering	✓	✓	✓	✓
5	Designing	✓	✓		✓
6	Implementation	✓	✓	✓	
7	Documentation	✓	✓	✓	✓

APPENDIX D

PO, PSO, PEO, AND CO RELEVANCE WITH PROJECT

CO-PO MAPPING SHEET

OUTCOME NO	DESCRIPTION
CO1	Develop problem formulation and design skills for solving real-world engineering problems.
CO2	Conduct literature surveys to analyze current research trends and develop analytical and presentation capabilities.
CO3	Gain expertise in software and hardware tools relevant to industry standards.
CO4	Foster innovative thinking and promote lifelong learning through research initiatives.
CO5	Enhance teamwork, presentation, and communication abilities.
CO6	Build a platform that enhances student employability.

SUMMARY OF CO MAPPING TO PROGRAM OUTCOMES

COs/POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PS1	PSO2
CO1	3	0	0	1	0	1	1	1	3	3	0	1	0	0
CO2	3	3	0	0	0	2	0	0	3	2	0	0	0	0
CO3	2	0	1	1	3	0	0	0	3	2	0	0	0	0
CO4	3	0	0	3	3	0	3	1	3	3	1	1	0	0
CO5	2	0	0	0	2	0	0	0	3	3	0	3	0	0
CO6	2	1	0	0	3	1	0	3	3	2	2	2	0	0
Overall Course	3	1	1	1	2	1	1	0	3	2	1	1	0	0

PROGRAM OUTCOMES (POs)

POs	PROGRAM OUTCOMES	RELEVANCE
PO1	Engineering Knowledge: Apply knowledge of mathematics, natural science, computing, engineering fundamentals and an engineering specialization as specified in WK1 to WK4 respectively to develop to the solution of complex engineering problems.	The research applies engineering knowledge, including machine learning, image processing, and neural networks, to address Medicinal Plant Image Processing.
PO2	Problem Analysis: Identify, formulate, review research literature and analyze complex engineering problems reaching substantiated conclusions with consideration for sustainable development.	The study identifies and analyzes the challenge of accurate Medicinal Plant Identification in Plant Images, considering false positives/negatives, and proposes substantiated conclusions through the CNN-based deep learning model.
PO3	Design/Development of Solutions: Design creative solutions for complex engineering problems and design/develop systems, components, or processes to meet identified needs with consideration for the public health and safety, whole-life cost, net zero carbon, culture, society, and environment as required.	The paper proposes a solution using CNN architecture for the early and accurate detection of brain tumors, incorporating deep learning for efficient medical diagnosis and patient safety.
PO4	Conduct Investigations of Complex Problems: Conduct investigations of complex engineering problems using research-based knowledge including design of experiments, modelling, analysis, and interpretation of data to provide valid conclusions.	The research includes experiments using datasets, model evaluation metrics (accuracy, sensitivity, specificity), and comparisons with other models, reflecting strong investigational methodology.
PO5	Engineering Tool Usage:Create, select and	The project employs modern IT

	apply appropriate techniques, resources, and modern engineering and IT tools, including prediction and modelling, recognizing their limitations to solve complex engineering problems.	tools such as Python, TensorFlow/Keras, and Plant image datasets, leveraging AI-based techniques and pre-trained networks like CNN for prediction and modeling.
PO6	The Engineer and the World: Analyze and evaluate societal and environmental aspects while solving complex engineering problems for their impact on sustainability with reference to economy, health, safety, legal framework, culture, and environment.	The application supports sustainable Plant Identification and helps in finding the medicinal plants and their uses in a better way.
PO7	Ethics: Apply ethical principles and commit to professional ethics, human values, diversity and inclusion; adhere to national and international laws.	The paper acknowledges ethical considerations in handling medical data and ensuring responsible use of AI in sensitive healthcare applications.
PO8	Individual and Collaborative Team Work: Function effectively as an individual, and as a member or leader in diverse/multidisciplinary teams.	The study involves interdisciplinary knowledge, combining engineering, computer science, and healthcare, requiring collaboration across multiple domains.
PO9	Communication: Communicate effectively and inclusively within the engineering community and society at large, such as being able to comprehend and write effective reports and design documentation, and make effective presentations considering cultural, language, and learning differences.	The research is well-documented and presented with clear methodology, results, and comparisons, effectively communicating complex ideas to both technical and medical audiences
PO10	Project Management and Finance: Apply knowledge and understanding of engineering management principles and economic	The development process reflects efficient use of computational resources and planning, which is

	decision-making and apply these to one's own work, as a member and leader in a team, and to manage projects in multidisciplinary environments.	vital for managing AI healthcare solutions within time and budget constraints.
PO11	Life-Long Learning: Recognize the need for, and have the preparation and ability for independent and life-long learning, adaptability to new and emerging technologies, and critical thinking in the broadest context of technological change.	The implementation of a cutting-edge architecture like CNN shows continuous learning and adaptation to emerging technologies in the evolving field of AI and healthcare.

PROGRAM SPECIFIC OUTCOME (PSOs)

PSOs	Program Specific Outcome	Relevance
PSO1	Students will be able to utilize core principles of Artificial Intelligence Engineering for the design, development and prototyping of AI Subsystems.	The research demonstrates the design and development of an AI-based subsystem using CNN, a deep convolutional neural network, for the detection of brain tumors. The prototype effectively processes Plant images and automates tumor classification, showcasing the application of AI engineering principles.
PSO2	Students will be able to employ acquired knowledge in data storage, data analytics, and Machine Intelligence to address and solve practical business challenges.	The study utilizes machine intelligence and deep learning techniques for medical image analysis, applying data preprocessing, model training, and performance evaluation to address a critical real-world healthcare problem—brain tumor detection—thus reflecting practical application of acquired AI and data analytics skills.

PROGRAMME EDUCATIONAL OBJECTIVES (PEOs)

PSOs	Program Specific Outcome	Relevance
PSO1	Students will be able to utilize core principles of Artificial Intelligence Engineering for the design, development and prototyping of AI Subsystems.	The research demonstrates the design and development of an AI-based subsystem using Convolutional Neural Networks (CNN) for the classification of medicinal plants. The prototype accurately processes leaf images and automates plant species identification, showcasing the application of AI engineering principles in a biological and agricultural context.
PSO2	Students will be able to employ acquired knowledge in data storage, data analytics, and Machine Intelligence to address and solve practical business challenges.	The study applies machine intelligence and deep learning techniques for plant image analysis, involving dataset collection, preprocessing, CNN model training, and evaluation. This approach addresses a real-world problem in the field of ethnobotany and agriculture, highlighting the practical use of AI and data analytics skills to support biodiversity and medicinal research.

COURSE OUTCOME (COs)

COs	Course Outcome	POs, PSOs, and PEOs Mapped
CO1	Develop problem formation and design skills for engineering and real-world problems	PO1, PO2, PO3, PSO1, PEO1
CO2	Collect and generate ideas through literature surveys on current research areas which help to analyze and formulate solutions.	PO2, PO3, PO5, PO6, PSO2, PEO2
CO3	Import knowledge of software & hardware to meet industry perspective needs and standards.	PO2, PO3, PO5, PO6, PSO2, PEO2
CO4	Create interest to research innovative ideas as lifelong learning.	PO11, PSO2, PEO2
CO5	Ability to work with a team, and enrich presentation and communication skills.	PO8, PO9, PO10, PEO3
CO6	Create a platform that makes students employable.	PO5, PO9, PO11, PSO2, PEO1

RELEVANCE TO POs

CO	PO	PI	Relevance
CO1	PO1,	1.2.1	Applied CNN-based classification using engineering fundamentals such as linear algebra, statistics, and optimization.
	PO2,	2.4.1	Defined the classification challenge as a complex engineering problem and analyzed Plant image data.
	PO3,	3.5.1	Designed a deep learning model architecture (CNN) to meet Plant Image requirements.
	PSO1,	-	Demonstrated the design and development of a real-time AI subsystem for Plant classification.
	PEO1	-	Applies domain knowledge in AI and technology to address real-world Plant Identification issues.
CO2	PO2,	2.6.4	Compared multiple AI techniques through literature and

			selected CNN as the best solution.
	PO3,	3.6.1	Transformed research findings into a viable engineering model using DL concepts.
	PO5,	5.4.2	Utilized machine learning tools and libraries for analysis and experimentation.
	PO6,	6.4.1	Assessed the societal and healthcare implications of the AI system’s performance.
	PSO2,	-	Used research and data analytics to create a practical application.
	PEO2	-	Encourages continued learning and research orientation in AI.
CO3	PO2,	5.5.1	Selected suitable tools (e.g., TensorFlow, Keras) and evaluated their performance and limitations.
	PO3,	7.3.1	Considered responsible AI development in Plant Classification, minimizing potential harm.
	PO5,	8.3.1	Maintained professional conduct while handling sensitive medical data.
	PO6,	9.4.2	Collaborated effectively on code integration and UI development.
	PSO2,	-	Showcases real-world prototyping of an AI application.
	PEO2	-	Equips students with tools and experience relevant to industry demands.
CO4	PO11,	11.3.2	Encourages independent learning by exploring and customizing pre-trained models.
	PSO2,	-	Integrates emerging tech with analytics for future-ready research.
	PEO2	-	Supports higher education and research with exposure to advanced architectures like CNN.
CO5	PO8,	8.3.1	Demonstrated ethical teamwork during model development and result validation.
	PO9,	9.5.2	The collaborative structure of the paper and co-authorship displays effective team engagement.
	PO10,	10.4.2	Documentation and presentation of the system was professional and structured.

	PEO3	-	Fosters leadership and ethical communication in technical environments.
CO6	PO5,	5.6.1	Validated the use of CNN and explained model training and evaluation processes.
	PO9,	9.5.1	Real-time deployment via Flask demonstrated practical communication and UI integration.
	PO11,	11.6.1	Identified project components and resources needed for real-world deployment.
	PSO2,	-	Used data science tools to create solutions for business and healthcare problems.
	PEO1	-	Builds employable skills through hands-on experience with industrial-grade AI systems.

APPENDIX E

PUBLICATIONS

Acceptance Letter of IJSREM Conference

Your Research Article Has Been Published – Celebrate Your Success! IJSREM Journal External Inbox ×

IJSREM Journal <editor@ijsrem.com> Mon, Apr 14, 4:22 PM (5 days ago)
to me

Dear Author,

We would like to congratulate you on the successful publication of your research paper in our journal! **International Journal of Scientific Research in Engineering and Management (IJSREM)** on **Volume 09, Issue 04 April 2025**.

Paper Title : **Revitalizing Ayurveda with Deep Learning: Automated Identification of Medicinal plants using CNN**

Your hard work has been noted and appreciated by us, and we thank you for showing interest in our journal. Your work is now visible to the world, and as an appreciation of your contribution we enclose with this email e-certificate of all the author(s) as a token from us.

DOI: 10.55041/IJSREM44506

For DOI verification, Click here https://www.doi.org/10.55041/IJSREM44506

Please login to https://www.doi.org and paste the DOI name into the text box.

We've also included a link through which you can download your published paper.

Publication Certificate 1

IJSREM e-Journal

ISSN: 2582-3930

ACCEPTANCE CERTIFICATE

Impact Factor: 8.586
DOI Prefix: 10.55041

INTERNATIONAL JOURNAL OF SCIENTIFIC RESEARCH IN ENGINEERING & MANAGEMENT (IJSREM)

An Open Access Scholarly Journal || Index in major Databases & Metadata

We are pleased to inform you that your manuscript titled

Revitalizing Ayurveda with Deep Learning: Automated Identification of Medicinal plants using CNN

has been ACCEPTED for publication in

INTERNATIONAL JOURNAL OF SCIENTIFIC RESEARCH IN ENGINEERING & MANAGEMENT (IJSREM)

VOLUME 09 ISSUE 04 APRIL- 2025

We are delighted to see your commitment & hardwork to share your research is being recognized. We look foreward to helping you with all of your publication needs. Thank you for choosing IJSREM!!

Crossref CiteFactor

Editor-in-chief
IJSREM

www.ijsrem.com e-mail: editor@ijsrem.com

Publication Certificate 2


DOI: 10.55041/IJSREM44506

IJSREM e-Journal

ISSN: 2582-3930
Impact Factor: 8.586

INTERNATIONAL JOURNAL OF SCIENTIFIC RESEARCH IN ENGINEERING & MANAGEMENT
An Open Access Scholarly Journal || Index in major Databases & Metadata

CERTIFICATE OF PUBLICATION

International Journal of Scientific Research in Engineering & Management is hereby awarding this certificate to

N. Deepak Sai

in recognization to the publication of paper titled

Revitalizing Ayurveda with Deep Learning: Automated Identification of Medicinal plants using CNN

published in IJSREM Journal on Volume 09 Issue 04 April, 2025

www.ijsrem.com

Editor-in-Chief
IJSREM Journal

e-mail: editor@ijsrem.com

Publication Certificate 3


DOI: 10.55041/IJSREM44506

IJSREM e-Journal

ISSN: 2582-3930
Impact Factor: 8.586

INTERNATIONAL JOURNAL OF SCIENTIFIC RESEARCH IN ENGINEERING & MANAGEMENT
An Open Access Scholarly Journal || Index in major Databases & Metadata

CERTIFICATE OF PUBLICATION

International Journal of Scientific Research in Engineering & Management is hereby awarding this certificate to

K. Pravallika

in recognization to the publication of paper titled

Revitalizing Ayurveda with Deep Learning: Automated Identification of Medicinal plants using CNN

published in IJSREM Journal on Volume 09 Issue 04 April, 2025

www.ijsrem.com

Editor-in-Chief
IJSREM Journal

e-mail: editor@ijsrem.com

Publication Certificate 4

DOI: 10.55041/IJSREM44506

IJSREM e-Journal

ISSN: 2582-3930
Impact Factor: 8.586

INTERNATIONAL JOURNAL OF SCIENTIFIC RESEARCH IN ENGINEERING & MANAGEMENT

An Open Access Scholarly Journal || Index in major Databases & Metadata

CERTIFICATE OF PUBLICATION

International Journal of Scientific Research in Engineering & Management is hereby awarding this certificate to

K. Varshini

in recognization to the publication of paper titled

Revitalizing Ayurveda with Deep Learning: Automated Identification of Medicinal plants using CNN

published in IJSREM Journal on Volume 09 Issue 04 April, 2025

www.ijsrem.com

Editor-in-Chief
IJSREM Journal

e-mail: editor@ijsrem.com

Publication Certificate 5

DOI: 10.55041/IJSREM44506

IJSREM e-Journal

ISSN: 2582-3930
Impact Factor: 8.586

INTERNATIONAL JOURNAL OF SCIENTIFIC RESEARCH IN ENGINEERING & MANAGEMENT

An Open Access Scholarly Journal || Index in major Databases & Metadata

CERTIFICATE OF PUBLICATION

International Journal of Scientific Research in Engineering & Management is hereby awarding this certificate to

M. Ajay

in recognization to the publication of paper titled

Revitalizing Ayurveda with Deep Learning: Automated Identification of Medicinal plants using CNN

published in IJSREM Journal on Volume 09 Issue 04 April, 2025

www.ijsrem.com

Editor-in-Chief
IJSREM Journal

e-mail: editor@ijsrem.com