**A PROJECT REPORT ON**

PLANT SPECIES RECOGNIZATION

Submitted in partial fulfilment of the requirement

For the award of the degree of

Bachelor of Technology

IN

## Computer Science and Engineering

Submitted by

GUNDAM ROHITH REDDY

(20H51A05E0)

PURUMALLA ARYA PATEL

(20H51A05F3)

D.VENKATA BHUVANESWAR REDDY

(20H51A05D9)

Under the guidance of

Mr Ranjith Kumar

(Associate Professor)



Department of Computer Science and Engineering

CMR COLLEGE OF ENGINEERING & TECHNOLOGY

(An Autonomous Institution, Approved by AICTE, Affiliated to JNTUH, NAAC ’A+’)

KANDLAKOYA, MEDCHAL ROAD, HYDERABAD - 501401.

2020- 2021

### CMR COLLEGE OF ENGINEERING & TECHNOLOGY

KANDLAKOYA, MEDCHAL ROAD, HYDERABAD – 501401

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

## 

## CERTIFICATE

This is to certify that the Project report “**Plant Species Recognization”** being submitted by G.Rohith Reddy (20H51A05E0), P.Arya Patel (20H51A05F3), D.V.Bhuvaneswar Reddy (20H51A05D9), in partial fulfilment for the award of Bachelor of Technology in Computer Science and Engineering record of bonafide work carried out his/her under my guidance and supervision.

e

The results embodies in this project report have not been submit

any other University or Institute for the award of any Degree.

Mr Ranjith Kumar Dr.S.Siva Skandha

Associate Professor Associate Professor and HOD

Dept. of CSE Dept. of CSE

2

2

## Acknowledgement

## 

## With great pleasure I want to take this opportunity to express my heartfelt gratitude to

## the people who helped in making this project work a grand success.

I am grateful to **Mr.Ranjith Kumar,** Associate Professor, Dept of Computer Science and Engineering for his valuable suggestions and guidance during the execution of this project work.

We would like to thank **Dr. S.Siva Skanda,**Head of the Department of Computer Science and Engineering, CMR College of Engineering and Technology, who is the major driving forces to complete my project work successfully.

We are very grateful to **Dr. Vijaya Kumar Koppula**, Dean-Academic, CMR College of Engineering and Technology, for his constant support and motivation in carrying out the project work successfully.

I am highly indebted to **Dr. V A Narayana, Principal** CMRCET for giving permission to carry out this project in a successful and fruitful way.

I would like to thank the Teaching & Non- teaching staff of Department of Computer Science and Engineering for their co-operation

Finally I express my sincere thanks to Mr. Ch. Gopal Reddy, Secretary, CMR Group of Institutions, for his continuous care. I sincerely acknowledge and thank all those who gave support directly and indirectly in completion of this project work.

G.Rohith Reddy – 20H51A05E0

P.Arya Patel -20H51A05F3

D.V.Bhuvaneswar reddy-20H51A05D9



**TABLE OF CONTENTS**

|  |  |  |
| --- | --- | --- |
| **CHAPTER NO.** | **TITLE** | **PAGE NO.** |
|  | **LIST OF FIGURES** | **5** |
|  | **ABSTRACT** | **6** |
| **1** | **INTRODUCTION** | **7** |
| **2** | **BACKGROUND WORK** | **8** |
| **3** | **PROPOSED SYSTEM**  **3.1 Introduction**  **3.2 Technology used**  **3.3 Importance of python**  **3.4 Existing System** | **9-10** |
| **4** | **DESIGNING**  **4.1 Proposed Architecture**  **4.2 Screenshots of code and database** | **11-21** |
| **5** | **RESULT AND DISCUSSION** | **22-26** |
| **6** | **CONCLUSION&FUTURE WORK** | **27** |
|  | **REFERENCES** | **28** |

4

**LIST OF FIGURES**

|  |  |  |
| --- | --- | --- |
| **FIGURE N0.** | **TITLE** | **PAGE NO.** |
| **4.1** | **PROPOSED ARCHITECTURE** | **11** |
| **4.2** | **USE CASE DIAGRAM** | **12** |
| **5.1**  **5.1.2**  **5.1.3**  **5.1.4**  **5.1.5** | **WEBSITE**  **REGISTER CREDENTIAL**  **LOGIN CREDENTIAL**  **CHOOSE FILE**  **OUTPUT OF LEAF** | **23**  **24**  **25**  **26**  **27** |



**ABSTRACT**

Plant plays an important role in agricultural, industrial, medicine, environmental and ecological protection. Recently, with global warming, biodiversity loss, rapid urban development and environmental damage, people have been seriously destroying the natural environments, which results in that a large number of plant species constantly dying and even dying out every year. It is essential to protect plant species. The first step of protecting plants is to recognise them and understand what they are and where they come from. But there are a large number of plant species that have been named on Earth, and many are still unknown yet, it is difficult to identify each species. To handle such huge information, develop a quick and efficient classification method has become significant research.

Plant species can be recognised by its leaf, flower, skin, fruit and seed, etc. Relatively speaking, using leaf to recognise plant species is very simple and convenient, and many leaf based plant species recognition methods have been proposed. In this paper, we mainly summarise the existing leaf-based plant species identification methods, including plant leaf characteristic, public databases, feature extraction-based methods, subspace learning based methods, sparse representation-based methods, and deep learning-based methods. The aim is to emphasise the importance of plant species identification, train people to know about plant species, and provide guidance and comprehensive study for the beginners in this field, in turn, to treasure and protect plant species.

6

CHAPTER-1

INTRODUCTION

Plant species recognition has received a great attention from the machine learning and

machine vision communities. There are many challenges that need to be overcome in order to have a reliable plant recognition method that could be used in serious industrial applications. The problem of plant species recognition can be seen as two separate problems, based on the goal that we want to achieve. The first problem can be defined as recognising the presence of a certain plant on an image (retrieval problem), and the second problem is finding a certain type of plant in an image and segmenting it from the image.

Plant Taxonomy is a science to separate plants into similar groups based on the

characteristics like colour of the flower, shape of the flower, leaf shape and form, fruits, bark of the stem etc. Plant recognition when done manually by specialised taxonomists, suffers from perceptual biases, cost of hiring of experts and shortage of experts. The process of manual taxonomy becomes time consuming and tedious when more and more images are added to the database. Automation of the recognition process can improve the time, efficiency, accuracy and cost associated with the recognition process. This has given rise to the demand of automatic tools for plant species recognition and classification. Major support system for the automation is availability of digitised databases with high resolution plant images annotated with species names and metadata like date and time, lat long information aided by high resolution cameras available on handheld devices. The current approaches for plant species identification utilise only leaf form and shape. The reason being that leaves due to their shape and size, have the advantage to be easily observed, captured and described and are less effected by seasonal changes. Some of the approaches use combination of leaf and bark or colour, shape and texture of the flowers to identify the species of the plant. The method uses content as well as the metadata associated with the image like to which part the image is associated and latitude and longitude of the place where image is captured We have used different techniques for different parts of the plant by extracting different features according to the part of the plant.



CHAPTER 2

BACKGROUND WORK

**2.1 Project Introduction:**

**2.1.1 What is “Plant Recognition”**

There are many challenges that need to be overcome in order to have a reliable plant recognition method that could be used in serious industrial applications. The problem of plant species recognition can be seen as two separate problems, based on the goal that we want to achieve. The first problem can be defined as recognising the presence of a certain plant on an image (retrieval problem), and the second problem is finding a certain type of plant in an image and segmenting it from the image.

**2.1.2 Why do we need** **“Plant Recognition”**

Plants recognition system is mostly built using image processing techniques where the leaf is considered the main organ in the full plant structure.

Leaf, which contains important information about the plant to which it belongs, is used for species identification as well as for variety identification.

CHAPTER 3

PROPOSED SYSTEM

**3.1 INTRODUCTION**

Instead, the image data set is collected whit different plant names then after that feature extraction

is done. Using deep learning we train to get new image and whenever new image is provided, we try finds new species of the plant. Thus, the prediction is done using Machine Learning

Proposed system features are:

1.User Interface - A UI is created to enable uploading of image(s) in which

the plant species are to be detected.

1. Reduced Human Errors - Using a machine drastically reduces chances of errors by human errors.

#### **3.2 Technologies Used**

1. Python

Python is a high-level, interpreted, interactive and object-oriented scripting language. Python is designed to be highly readable. It uses English keywords frequently where as other languages use punctuation, and it has fewer syntactical constructions than other languages.

History of Python

Python was developed by Guido van Rossum in the late eighties and early nineties at the National Research Institute for Mathematics and Computer Science in the

Netherlands.

Python is derived from many other languages, including ABC, Modulo-3, C, C++, Algol-68, SmallTalk, and Unix shell and other scripting languages.

Python is copyrighted. Like Perl, Python source code is now available under the GNU

General Public License (GPL). Python is now maintained by a core development team at the institute, although Guido van Rossum still holds a vital role in directing its progress.

**3.3 Importance of Python**

* Python is Interpreted − Python is processed at runtime by the interpreter. You do not need to compile your program before executing it. This is similar to PERL and PHP.

* Python is Interactive − You can actually sit at a Python prompt and interact with the interpreter directly to write your programs.

* Python is Object-Oriented − Python supports Object-Oriented style or technique of programming that encapsulates code within objects.

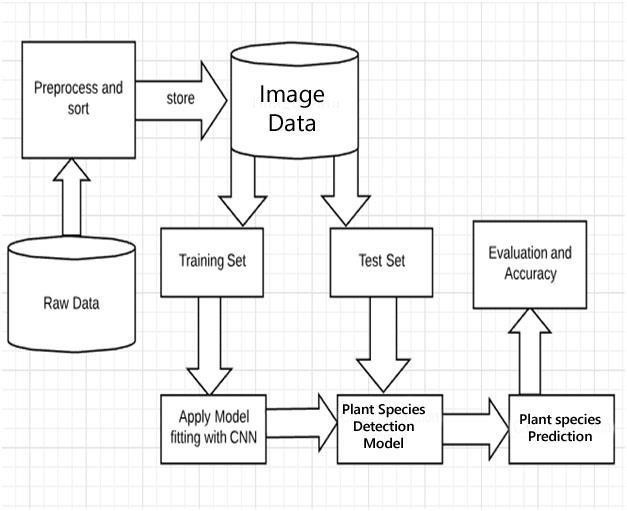
* Python is a Beginner's Language − Python is a great language for the beginner-level programmers and supports the development of a wide range of applications from simple text processing to WWW browsers to games

3.4 EXISTING SYSTEM

Plant Taxonomy is a science to separate plants into similar groups based on the characteristics like colour of the flower, shape of the flower, leaf shape and form, fruits, bark of the stem etc. Plant recognition when done manually by specialised taxonomists, suffers from perceptual biases, cost of hiring of experts and shortage of experts. The process of manual taxonomy becomes time consuming and tedious when more and more images are added to the database. Automation of the recognition process can improve the time, efficiency, accuracy and cost associated with the recognition process. This has given rise to the demand of automatic tools for plant species recognition and classification. Major support system for the automation is availability of digitised databases with high.

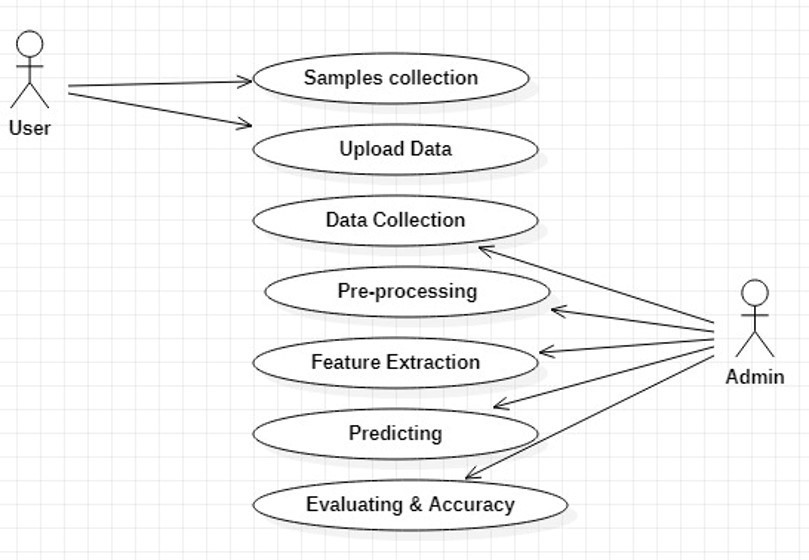
DESIGNING

**4.1 Proposed Architecture**



4.1 the above figure represents the architecture of plant species recognization

**4.2 Use case diagram**



**4.3 code**

# Libraries

from django.shortcuts import render,redirect

from django.http import HttpResponse

import pandas as pd

import numpy as np

import matplotlib.pyplot as plt

from sklearn.model\_selection import train\_test\_split

from sklearn.feature\_extraction.text import TfidfVectorizer

import itertools

from sklearn.naive\_bayes import MultinomialNB

from sklearn import metrics

from sklearn.linear\_model import PassiveAggressiveClassifier

import os

import seaborn as sns

from sklearn.linear\_model import LogisticRegression

from sklearn.svm import SVC

from sklearn.tree import DecisionTreeClassifier

from sklearn.neighbors import KNeighborsClassifier

from sklearn.model\_selection import train\_test\_split

from sklearn.metrics import confusion\_matrix

# Input data files are available in the "../input/" directory.

# For example, running this (by clicking run or pressing Shift+Enter) will list the files in the input directory

from fakenews.models import User

################ Home #################

def home(request):

return render(request,'home1.html')

def login(request):

return render(request,'loginform.html')

def loginCheck(request):

if request.method == 'POST':

firstname = request.POST.get('username')

password = request.POST.get('email')

request.session['firstname'] = firstname

print(firstname)

print(password)

user\_object=User.objects.get(firstname=firstname,password=password)

print('--------------------')

print(user\_object)

try:

print('hi')

user\_object=User.objects.get(firstname=firstname,password=password)

print(user\_object)

if user\_object is not None:

print('hiiiiiiii')

request.session['useremail'] = user\_object.email

return redirect('home')

except:

#user\_object = None

print('hello')

return redirect('login')

return render(request,'home.html')

def logout(request):

return render(request,'index.html')

def reg(request):

return render(request,'register.html')

######## SVM ######

def save(request):

if request.method == 'POST':

print('printtttttttttttttttttttttttttttttttt')

print('checkkkkkkkkkkkkkkkkk')

username= request.POST.get('username')

password= request.POST.get('password')

address= request.POST.get('address')

email= request.POST.get('email')

age= request.POST.get('age')

gender= request.POST.get('gender')

phone= request.POST.get('phone')

user=User()

user.firstname= request.POST.get('username')

user.password= request.POST.get('password')

user.address= request.POST.get('address')

user.email= request.POST.get('email')

user.age= request.POST.get('age')

user.gender= request.POST.get('gender')

user.phone= request.POST.get('phone')

user.save()

return render(request,'loginform.html')

return render(request,'loginform.html')

######## SVM ######

def nvb(request):

return render(request,'pacweb1.html')

def pac(request):

return render(request,'result.html')

def svm(request):

return render(request,'acc1.html')

def dec(request):

return render(request,'acc1.html')

def randomf(request):

return render(request,'acc1.html')

def mnb(request):

return render(request,'acc1.html')

def graph(request):

if request.method == 'POST' and request.FILES['myfile']:

# Importing libraries

import os

import pandas as pd

import numpy as np

import matplotlib.pyplot as plt

from matplotlib.image import imread

import seaborn as sns

import random

from PIL import Image

from sklearn.model\_selection import train\_test\_split

from tensorflow.keras.utils import to\_categorical

import tensorflow as tf

from tensorflow.keras.models import Sequential, model\_from\_json

from tensorflow.keras.layers import Dense, Flatten, Dropout, Conv2D, MaxPool2D

myfile = request.FILES['myfile']

file1=myfile.name

######

##########################

################################

import sys,os

print('\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*')

print(os.path.join(os.getcwd(), 'img', file1))

print('\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*')

import os

dirname = os.path.dirname(\_file\_)

filename = os.path.join(dirname, myfile.name)

print(filename)

global model

from keras.models import model\_from\_json

dim1 = []

dim2 = []

for i in range(0,3):

labels = 'C:/Users/bhuva/Desktop/leaf\_detect/images' + '/{0}'.format(i)

image\_path = os.listdir(labels)

for x in image\_path:

img = imread(labels + '/' + x)

dim1.append(img.shape[0])

dim2.append(img.shape[1])

print("Dimension 1 Mean : ",np.mean(dim1), " Dimension 2 Mean : ",np.mean(dim2))

images = []

label\_id = []

for i in range(3):

labels = 'C:/Users/bhuva/Desktop/leaf\_detect/images' + '/{0}'.format(i)

image\_path = os.listdir(labels)

for x in image\_path:

img = Image.open(labels + '/' + x)

img = img.resize((50,50))

img = np.array(img)

images.append(img)

label\_id.append(i)

images = np.array(images)

images = images/255

label\_id = np.array(label\_id)

label\_id.shape

images.shape

label\_counts = pd.DataFrame(label\_id).value\_counts()

label\_counts.head()

x\_train, x\_val, y\_train, y\_val = train\_test\_split(images, label\_id , test\_size = 0.2, random\_state = 42)

y\_train\_cat = to\_categorical(y\_train)

y\_val\_cat = to\_categorical(y\_val)

model = Sequential()

model.add(Conv2D(filters = 64, kernel\_size = (3,3), input\_shape = x\_train.shape[1:], activation = 'relu', padding = 'same'))

model.add(MaxPool2D(pool\_size=(2,2)))

model.add(Dropout(0.5))

model.add(Conv2D(filters = 64, kernel\_size = (3,3), activation = 'relu'))

model.add(MaxPool2D(pool\_size=(2,2)))

model.add(Dropout(0.5))

model.add(Flatten())

model.add(Dense(128, activation = 'relu'))

model.add(Dropout(0.5))

model.add(Dense(4, activation = 'softmax'))

model.compile(loss = 'sparse\_categorical\_crossentropy', optimizer = 'adam', metrics = ['accuracy'])

model.summary()

model.fit(x\_train, y\_train, epochs = 50, batch\_size = 128, validation\_data = (x\_val, y\_val), verbose = 2)

model\_json = model.to\_json()

test\_path1 = 'C:/Users/bhuva/Desktop/leaf\_detect/test\_images/'+file1

img = Image.open(test\_path1)

img = img.resize((50,50))

img = np.array(img)

im2arr = img.reshape(1,50,50,3)

test = np.asarray(im2arr)

test = test.astype('float32')

test = test/255

y\_pred1 = model.predict\_classes(test);

print(y\_pred1)

testimg\_path="C:/Users/bhuva/Desktop/leaf\_detect/test\_images/"+file1

if y\_pred1==0:

import cv2

from imutils import paths

import imutils

imagedisplay = cv2.imread(test\_path1)

orig = imagedisplay.copy()

output = imutils.resize(orig, width=400)

cv2.putText(output,"Pepper\_bell\_\_healthy",(10,25), cv2.FONT\_HERSHEY\_SIMPLEX,0.7, (0, 255, 0), 2)

cv2.imshow("Predicted Image Result", output)

cv2.waitKey(0)

return render(request, 'home1.html')

elif y\_pred1==1:

import cv2

from imutils import paths

import imutils

imagedisplay = cv2.imread(test\_path1)

orig = imagedisplay.copy()

output = imutils.resize(orig, width=400)

cv2.putText(output, "Potato\_\_\_healthy", (10, 25), cv2.FONT\_HERSHEY\_SIMPLEX,0.7, (0, 255, 0), 2)

cv2.imshow("Predicted Image Result", output)

cv2.waitKey(0)

elif y\_pred1==2:

import cv2

from imutils import paths

import imutils

imagedisplay = cv2.imread(test\_path1)

orig = imagedisplay.copy()

output = imutils.resize(orig, width=400)

cv2.putText(output, "Tomato\_healthy", (10, 25), cv2.FONT\_HERSHEY\_SIMPLEX,0.7, (0, 255, 0), 2)

cv2.imshow("Predicted Image Result", output)

cv2.waitKey(0)

return render(request, 'home1.html')

return render(request,'home1.html')

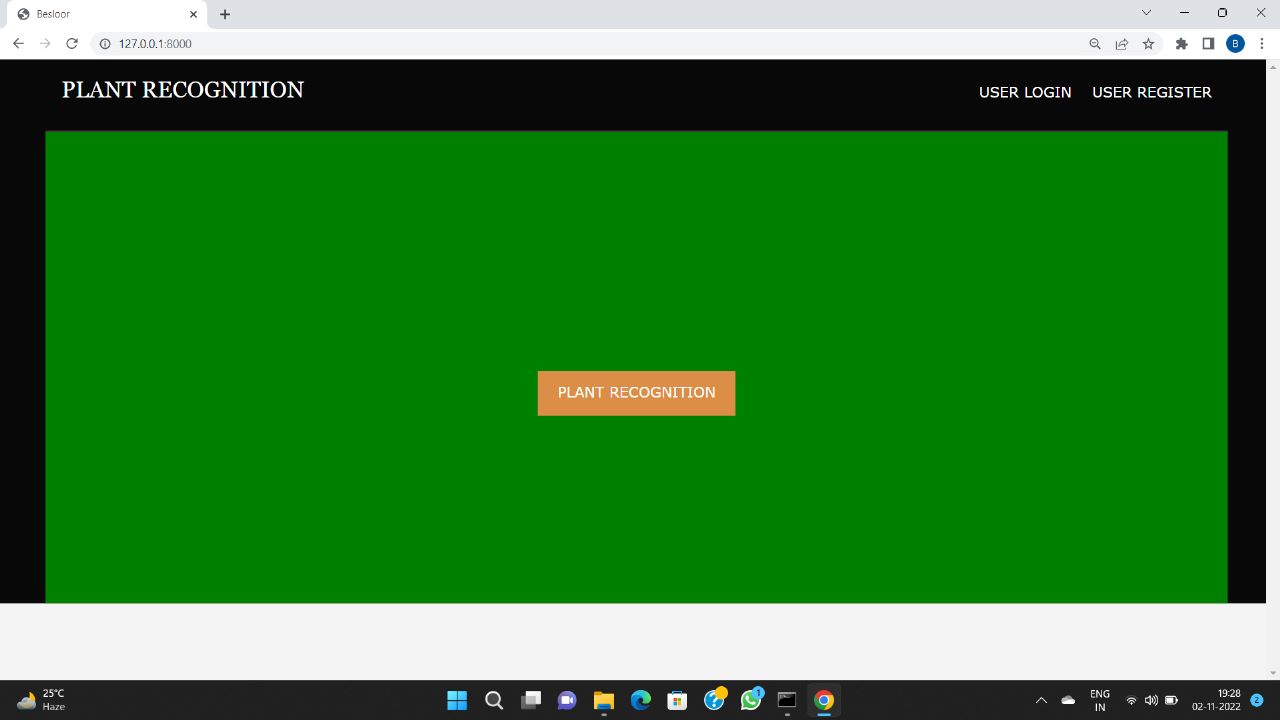
def accuracy(request):

return render(request,'index.html')

#### CHAPTER 5

RESULTS AND DISCUSSION

The Pictures shown below are executed process of the project, that’s the plant species recognization



Screenshots 5.1- Website

#### 

#### Screenshot 5.1.2- Register credentials

#### 

#### Screenshot 5.1.3- Login Credential

#### 

#### Screenshot 5.1.4- Choose file

#### 

#### Screenshot 5.1.5- output of leaf

#### CHAPTER 6

CONCLUSION AND FUTURE WORK

**CONCLUSION:**

Various techniques were used to predict the plant species in a metastatic image, based on visual observation and human experience, these predictions were often not very accurate.

However, in recent years, with the advancement in technology, it has been possible to study metastatic images correctly using machine learning techniques namely Convolution Neural Networks and NasNet.

In our project, we implemented these algorithms to predict the plant species by training the machine using a large dataset of collected metastatic scans and determined the roc curve for the best working algorithm.

**FUTURE WORK:**

Instead, the image data set is collected whit different plant names then after that feature extraction is done. Using deep learning we train to get new image and whenever new image is provided, we try finds new species of the plant. Thus, the prediction is done using Machine Learning.

#### 

#### REFERENCES

#### **REFERENCES**

1. https://www.analyticsvidhya.com/blog/2021/05/convolutional-neural-networksunderstand-the-basics/
2. https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neuralnetworks-the-eli5-way-3bd2b1164a53
3. https://ieeexplore.ieee.org/document/7411268