INSECT PEST CLASSIFICATION AND RECOMMENDATION OF PESTICIDES USING RANDOM FOREST ALGORITHM

A PROJECT REPORT

submitted by

UKESHWARN G   2116210701295

VASIKARAN G   2116210701305

VISHWA N   2116210701315

in partial fulfillment for the award of the

degree of

BACHELOR OF ENGINEERING

in

COMPUTER SCIENCE AND ENGINEERING

RAJALAKSHMI ENGINEERING COLLEGE

ANNA UNIVERSITY, CHENNAI

MAY 2024

RAJALAKSHMI ENGINEERING COLLEGE, CHENNAI

BONAFIDE CERTIFICATE

Certified that this Project report titled **"INSECT PEST CLASSIFICATION AND RECOMMENDATION OF PESTICIDES USING RANDOM FOREST ALGORITHM"** is the bonafide work of **"UKESHWARAN G -2116210701295, VASIKARAN G -2116210701305, VISHWA N -2116210701315"** who carried out the work under my supervision. Certified further that to the best of my knowledge the work reported herein does not form part of any other thesis or dissertation on the basis of which a degree or award was conferred on an earlier occasion on this or any other candidate.

SIGNATURE

Dr.P.SHANMUGAM,M.Tech.,Ph.D,

PROJECT COORDINATOR

Associate Professor

Department of Computer Science and Engineering

Rajalakshmi Engineering College

Chennai - 602 105

Submitted to Project Viva-Voce Examination held on ________________

Internal Examiner **External Examiner**

ABSTRACT

In agricultural ecosystems, the management of insect pests is crucial for maximizing crop yield and quality. Accurate identification and classification of pest species are essential steps in devising effective pest control strategies. In this project, we propose an approach leveraging the Random Forest algorithm for the classification of insect pests. Random Forest is a powerful machine learning technique capable of handling large datasets and complex feature interactions. We integrate our classification model with a recommendation system for selecting appropriate pesticides. By considering the identified pest species, the recommendation system suggests the most suitable pesticides, thereby aiding farmers in making decisions for pest management while minimizing environmental impact. This project offers a comprehensive solution for insect pest management, combining classification with pesticide recommendations. By harnessing the capabilities of machine learning, farmers can enhance their pest control practices, leading to improved crop productivity and sustainability. The proposed approach also contributes to the reduction of pesticide usage and associated risks, promoting more eco-friendly and efficient agricultural practices.

ACKNOWLEDGMENT

We thank the almighty god for the successful completion of the project. Our sincere thanks to our chairman **Mr. S. MEGANATHAN,B.E., F.I.E.,** for his sincere endeavor in educating us in his premier institution. We would like to express our deep gratitude and sincere thanks to our beloved Chairperson **Dr.(Mrs).THANGAM MEGANATHAN,Ph.D.,**for her enthusiastic motivation which inspired us a lot in completing this project and we extend our gratitude to our Vice Chairman **Mr.ABHAY SHANKAR MEGANATHAN,B.E., M.S.,** for providing us with the requisite infrastructure.

We also express our sincere and warmful gratitude to our college Principal **Dr. S.N.MURUGESAN,M.E.,Ph.D.**,and **Dr. P. KUMAR,M.E.,Ph.D.**, **Director of Computing and Information Science** and **Head Of the Department of Computer Science and Engineering** and **Project coordinator Dr.P.SHANMUGAM,M.Tech.,Ph.D.**,for his encouragement and guiding us throughout project towards successful completion of project and to our parents, friends, all faculty members,supporting staff for direct and indirect involvement in successful completion of project for their encouragement and support.

UKESHWARAN G
VASIKARAN G
VISHWA N

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

In agriculture the major difficulty faced by farmers is pest control and management. In addition to reduction in crop yield these pest also affects the quality of the agricultural products which leads to reduction in the price of the crops products because of this farmers suffers significant loss in their revenue. In Traditional pest control process the insect is identified manually which requires Specific knowledge and since the insects are identified manually it is a time consuming process and also there is a high chances of misclassification and using incorrect pesticides not only affects the crop but also damages the soil and also increases the production cost. Since traditional method requires a person who has sufficient knowledge to identify the type of pest and to suggest pesticides this leads to increase in the price of the crop production. In most cases farmers identify the type of pest based on their prior experience and knowledge which may leads to incorrect pest management decisions and Because of using wrong pesticides the pest might not be exterminated and also may leads to severe crop damage and reduction in soil quality which increases the price of crop production. Our project tries to overcome the disadvantages in traditional insect pest identification by using machine learning. By this project the human error which may occur in manual identification of pest can be solved. It also reduces the production cost by accurately identifying the insect pest and suggesting the most suitable pesticide for the identified type of pest.

1.1 PROBLEM STATEMENT

The presence of insect pests poses a significant threat to crop yields and quality. Pest infestations pose a significant threat to crops, leading to substantial economic losses. Traditional pest control methods are often inefficient and can result in the excessive use of pesticides, which harms the environment, human health, and crop yield. Agriculture production decreases as the insects feed on the crops. The purpose of this project is to provide a way to classify insect pest using Convolutional neural network and Random forest algorithm and recommend suitable pesticides to farmers.

1.2 SCOPE OF THE WORK

The Insect Pest Classification and Recommendation of Pesticides using Random Forest Algorithm tries to overcomes the difficulties in traditional insect pest classification methods such as human error which may occur in manual identification of pest which leads to misclassification of insect pest and incorrect pesticide usage. To solve this problem our project uses CNN and Random forest algorithm to classify the insect pest type and suggesting suitable pesticides thus helping farmers to reduce cost of crop production and increase the crop yield. Economically, it benefits farmers by decreasing crop losses and minimizing costs associated with excessive pesticide use. Utilizing cloud computing resources will allow for scalable implementation across diverse agricultural settings, from small farms to large agribusinesses.

1.3 AIM AND OBJECTIVES OF THE PROJECT

The primary objective of this project is to help farmers in making effective pest management strategies by providing user friendly interface for the users to identify insect pest and suggest suitable pesticides thus leading to efficient pest management and proper use of pesticide to improve crop productivity. Convolutional Neural Network is used to extract features from the insect dataset. The features extracted from the dataset is used to train the Random Forest Algorithm where each decision tree is trained on a part of the dataset to classify the type of insect pest and suggest the most suitable pesticide.

Random Forest Algorithm is used to classify the insect pest and convolutional neural network is used only for feature extraction from the image dataset. Therefore the overall computational time required for the system to classify insect pest is reduced. The project aims to provide farmers with a tools for pest management, for a more safer crop production and addressing one of the most critical challenges in modern agriculture.

1.4 RESOURCES

The resources required for the proposed solution includes a high performance computer with GPU for training and testing the deep learning model. Python programming language along with libraries such as pandas, scikit-learn, numpy, and matplotlib for data manipulation, model training, and evaluation. Deep learning frameworks such as TensorFlow, PyTorch, or Keras for developing and training convolutional neural networks (CNNs). Image processing libraries like OpenCV for preprocessing and feature extraction from plant images. Cloud services to deploy and host the trained model through web to ensure easily accessibility without the constraints of local storage.

1.5 MOTIVATION

The motivation behind the project insect pest classification and pesticide recommendation using the Random Forest algorithm is to improve agricultural productivity and sustainability. This project seeks to harness machine learning to provide a pest management solution. The Random Forest algorithm, will be used to classify pests based on various features which are extracted from the image dataset using CNN. Additionally, it will recommend effective pesticides, promoting environmentally friendly pest control practices. This approach supports sustainable agriculture by reducing the impact of pesticides, protecting crops, and ensuring safer crop production.

CHAPTER 2

2.1 LITERATURE SURVEY

“Large-Scale Insect Pest Image Classification by Thanh-Nghi Doan” [1] presents a novel method for large-scale insect pest image classification by combining fine-tuning EfficientNets and Power Mean Support Vector Machine (SVM). First, EfficientNet models are fine-tuned and re-trained on new insect pest image datasets. The retrieved features from EfficientNet models are then utilized to create a machine learning classifier.

“A Large-Scale Benchmark Dataset for Insect Pest Recognition by Rajak”[2],talks about crop prediction using various learners like SVM used as a classifier, Naive Bayes, Multilayer perceptron (ANN) and lastly Random Forest. The parameters used for crop prediction are: pH, depth, water holding capacity, drainage, erosion.

“Position dependent identification for field moth images using deep learning, by Chenglu Wen”[3] collects a large-scale dataset for insect pest identification called IP102, which contains over 75, 000 photographs of 102 insects. In comparison to previous datasets, the IP102 complies with a number of features of insect pest distribution in real-world settings . Meanwhile, they use the dataset to test certain cutting-edge recognition techniques. The findings show that existing handcrafted feature methods and deep feature methods are insufficient for pest identification.

“Pest detection and recognition using k-means clustering algorithm by Phil Birch, Rupert C D”[4] Using a machine learning and insect pest detection algorithm, various insect datasets were identified and detected, and the results were correlated. ANN, SVM, KNN, Naive Bayes, and the CNN model were used to test classification accuracy between different machine learning techniques. According to the findings, the CNN model has the best classification precision of 91.5 percent and 90 percent for 9 and 24 insect groups, respectively, from the Wang and Xie datasets.

“A study on various data mining techniques for crop yield prediction by Gandge, Yogesh, and Sandhya”[5] talks about Attribute selection, Multiple Linear Regression, Decision Tree using ID3, SVM, Neural Networks, C4.5, K-means and KNN. The proposed system consists of firstly Selection of agricultural field then Selection of crop previously planted, it takes input from user, preprocesses it, then in backend there is attribute selection followed by classification algorithm on data and then crop is recommended.

"Pest identification in images using SVM classifier by Amsini Ram"[6] describes a tool for automatically tracking pests using photographs from traps. A sliding window-based detection pipeline is proposed, in which a convolutional neural network is applied to image patches at various locations to decrease the success of possessing a particular pest type. To generate the final detections, image patches are filtered using non-maximum suppression and thresholding based on their positions and related confidences.

"Classification of Medicinal Plants Leaves Using Deep Learning Technique: A Review,2022, **by H. Chanyal, R.K. Yadav, and D.K.J. Saini"**[7] conduct a comprehensive review of the application of deep learning techniques for the classification of medicinal plant leaves. Firstly, it may explore the existing methodologies and technologies employed in the classification of medicinal plants, ranging from traditional taxonomical approaches to modern computational methods. The work highlights the advantages of deep learning over traditional image processing methods, particularly in handling the variability in leaf shapes, sizes, and textures.

"Crop Recommendation System to Maximize Crop Yield using Machine Learning Technique by Rajak, Rohit Kumar "[8] talks about crop prediction using various learners like SVM used as a classifier, Naive Bayes, Multilayer perceptron (ANN) and lastly Random Forest. The parameters used for crop prediction are: pH, depth, water holding capacity, drainage, erosion.

“Herbal Leaves Classification Based on Leaf Image Using CNN Architecture Model VGG16,202 by UN Oktaviana, GW Wicaksono, and AE Minarno's”[9] the authors undertake a significant endeavor in the realm of botanical science and computer vision. The proposed idea revolves around the classification of herbal leaves through the utilization of a convolutional neural network (CNN) architecture known as VGG16. The literature survey within this paper is likely to delve into several crucial domains. The VGG16 model demonstrated significant potential in accurately identifying herbal plants through image classification, suggesting its utility in botanical and pharmacological applications (EUDL).

“A Review on Extreme Learning Machine by J. Wang, S. Lu, S.H. Wang, and Y.D. Zhang”[10] provide an extensive examination of the Extreme Learning Machine (ELM) algorithm. Initially, it may delve into existing machine learning algorithms and techniques, ranging from traditional models like support vector machines and neural networks to more recent advancements such as deep learning architectures. This survey would likely underscore the limitations and advantages of each method, highlighting the increasing interest in ELM as an alternative approach for efficient and scalable learning tasks.

“Iris Features Extraction and Recognition Based on the Scale Invariant Feature Transform (SIFT) by M.A. Taha, H.M. Ahmed, and S.O. Husain” [11] undertake a detailed exploration of utilizing the Scale Invariant Feature Transform (SIFT) algorithm for iris feature extraction and recognition.Initially, it may delve into existing methodologies and technologies utilized for iris recognition, ranging from traditional techniques like template matching to more recent advancements in computer vision and machine learning.By situating ELM within the broader landscape of machine learning techniques, the review underscores its potential for efficient and scalable solutions in complex real-world problems.

"Crop Recommendation System for Precision Agriculture by Dighe, Deepti”[12] reviewed CHAID, KNN, K-means, Decision Tree, Neural Network, Naïve Bayes, C4.5, LAD, IBK and SVM algorithms and generated rules for recommendation system. Considering various factors like pH level of soil, month of cultivation, weather in the region, temperature, type of soil, etc. factors were considered to select maximum likely crops for plantation.

“A Recommendation System for Farmers by Mokarrama, Miftahul Jannat, and Mohammad Shamsul Arefin”[13] discussed Location Detection, Data analysis and storage, Similar location detection and Recommendation generation module. Physiographic database, Thermal zone database, Crop growing period database, crop production rate database and seasonal crop database were used to get the final crop.

“A study on various data mining techniques for crop yield prediction by Gandge, Yogesh, and Sandhya”[14], talks about Attribute selection, Multiple Linear Regression, Decision Tree using ID3, SVM, Neural Networks, C4.5, K-means and KNN. The proposed system consists of firstly Selection of agricultural field then Selection of crop previously planted, it takes input from user, preprocesses it, then in backend there is attribute selection followed by classification algorithm on data and then crop is recommended.

“Plant Disease and Pest Detection Using Deep Learning-Based Features by M. Türkoğlu and D. Hanbay”[15] present a comprehensive exploration of the application of deep learning techniques for the detection of plant diseases and pests. The literature survey within this paper likely delves into several key areas. Firstly, it may explore existing methodologies and technologies utilized for plant disease and pest detection, ranging from traditional visual inspection methods to more recent computer vision approaches. The review may discuss different approaches to data preprocessing, including image augmentation, normalization.

2.2 PROPOSED SYSTEM

DATASET:

Our approach involves a methodology that integrates Convolutional Neural Networks (CNNs) for feature extraction and the Random Forest algorithm for classification and recommendation tasks. This system is designed to identify insect pests and suggest appropriate pesticides.The initial step involves gathering a comprehensive dataset that includes images of various insect pests.

MODEL ARCHITECTURE:

Once the dataset is collected, CNNs are employed to extract meaningful features from the images. CNNs are particularly effective for this task due to their ability to automatically identify and learn hierarchical patterns in image data. The network is trained on a labeled dataset of pest images, allowing it to learn distinctive features such as shape, texture, and color, which are crucial for accurate pest identification.The dataset extracted by the CNN is used to train a Random Forest classifier. Random Forest is chosen for its robustness and ability to handle high-dimensional data. During training, the model learns to differentiate between various pest species based on the provided features. After training, the Random Forest model is deployed to classify new pest images. When a new image is input into the system, the CNN first extracts its features, which are then passed to the Random Forest classifier.

The classifier processes these features and assigns a label corresponding to the identified pest species.The system is implemented as a user-friendly application, allowing farmers to upload pest images and receive instant classifications and pesticide recommendations.

TRAINING AND TESTING:

The training and testing process begins with splitting the dataset into training and testing subsets, typically using an 80-20 split. This ensures that the model can be evaluated on unseen data, providing a realistic measure of its performance. Once training is complete, the model is validated on the testing subset to assess its generalization capability.To further understand the model's performance, loss and accuracy graphs are plotted over the training epochs. These graphs illustrate the model's learning process, showing how the loss decreases and the accuracy improves over time. By carefully monitoring these metrics and graphs, we ensure that the model is not only accurate but also generalizes well to new, unseen data, making it a reliable tool for pest classification and pesticide recommendation.

CHAPTER 3

SYSTEM DESIGN

3.1 GENERAL

In this section, we would like to show how the general outline of how all the components end up working when organized and arranged together. It is further represented in the form of a flow chart below.

3.2 SYSTEM ARCHITECTURE DIAGRAM

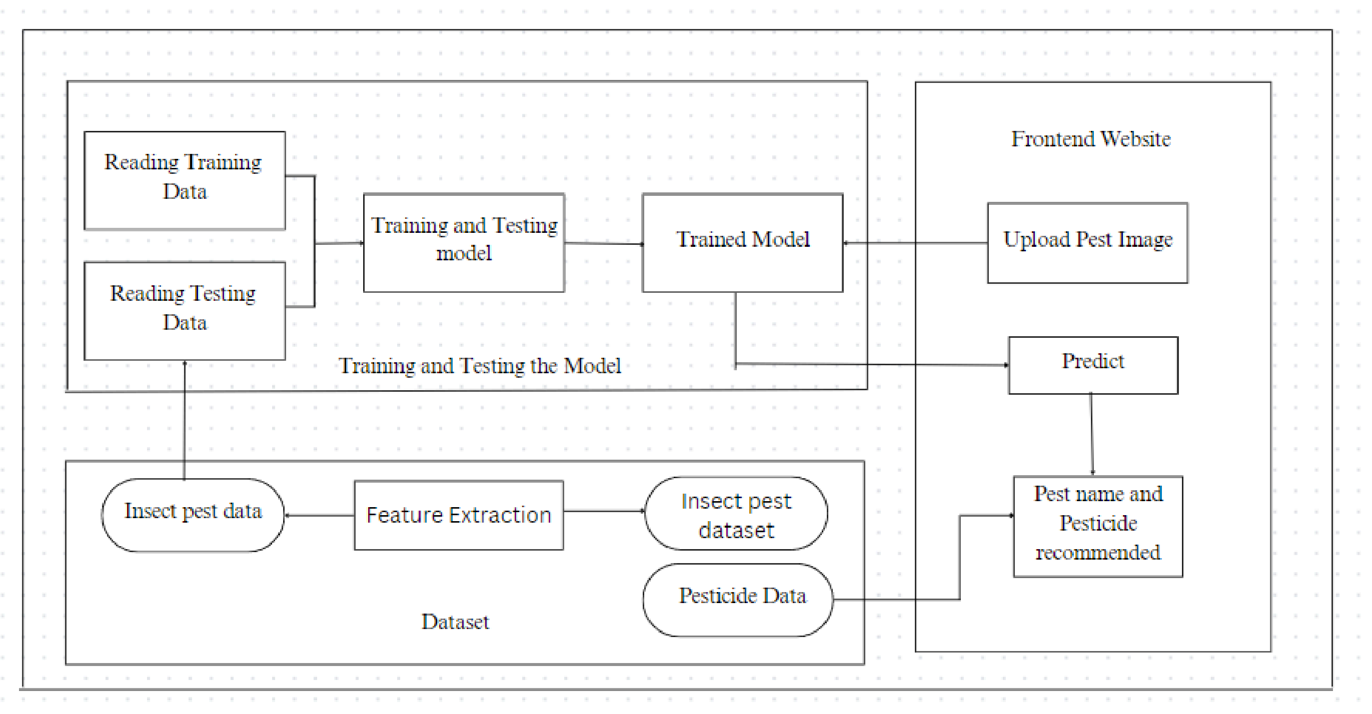

Fig 3.2.1: System Architecture

3.3 DEVELOPMENTAL ENVIRONMENT

3.3.1 HARDWARE REQUIREMENTS

The hardware requirements may serve as the basis for a contract for the system's implementation. It should therefore be a complete and consistent specification of the entire system. It is generally used by software engineers as the starting point for the system design.

Table 3.3.1 Hardware Requirements

COMPONENTS	SPECIFICATION
PROCESSOR	Intel Core i5
RAM	8 GB RAM
GPU	NVIDIA GeForce GTX 1650
MONITOR	15" COLOR
HARD DISK	512 GB
PROCESSOR SPEED	MINIMUM 1.1 GHz

3.3.2 SOFTWARE REQUIREMENTS

The software requirements document is the specifications of the system. It should include both a definition and a specification of requirements. It is a set of what the system should rather be doing than focus on how it should be done. The software requirements provide a basis for creating the software requirements specification.The software requirements are description of features and functionalities of the target system. Requirements convey the expectations of users from the software product.

Table 3.3.2 Software Requirements

S.NO	REQUIREMENT
1	Jupyter Notebook
2	StreamLit API
3	TensorFlow
4	MongoDB
5	Blockchain

3.4 DESIGN OF THE ENTIRE SYSTEM:

3.4.1 SEQUENCE DIAGRAM:

A sequence diagram simply depicts the interaction between the objects in a sequential order.An sequence diagram is used to show the interactive behavior of a system.The sequence diagram for medicinal plants identification with supply chain integrity is attached in the below figure 3.4.1.

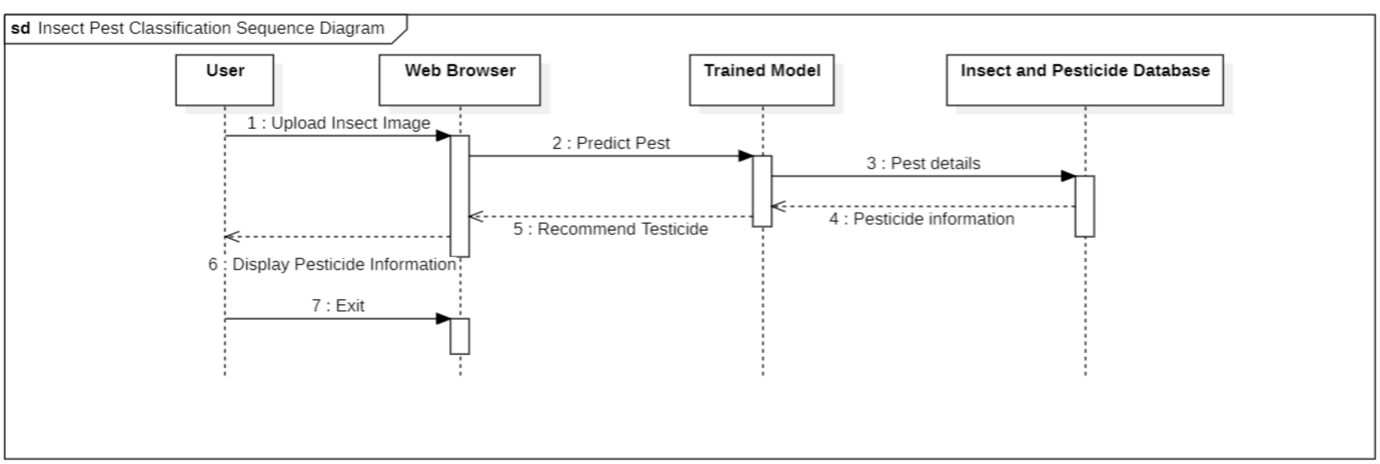

Fig 3.4.1: Sequence Diagram

CHAPTER 4

PROJECT DESCRIPTION

4.1 METHODOLOGY

There are many important elements in the methodology of classifing insect pest through random forest algorithm and recommending pesticides The methodology data collection which is gathering data on various types of insect pests that poses threat to farmers and collect information on different pesticides, focusing on their effectiveness against specific pests. Data preprocessing by which removing any irrelevant or low-quality photos from the dataset. Preprocessing procedures such as image normalization, scaling, and augmentation can help improve the dataset's quality and diversity. Model Selection and Training in which Convolutional neural networks (CNNs) is used to extract various features from the insect pest images and Random forest algorithm is trained on this data and used to classify the insect pest. Recommending suitable pesticide to the farmers which are most effective by using predicted type of the insect pest. Designing a user interface to upload insect image which is user friendly and display the results of the classification and th e recommended pesticide to the user. Create a user-friendly application or system that provides users with easy access.

4.2 MODULE DESCRIPTION

The Insect Pest Classification and Recommendation of Pesticide Using Random Forest Algorithm is divided into three major sections each serving a distinct purpose within System. The Deep Learning Model Training Module selects relevant models, such as CNNs, and use them on the labeled dataset to extract features from the insect pest dataset. The Pest Identification module conducts insect pest classification by analyzing user-provided uploaded images, utilizing a trained Random Forest classifier to identify pests and present the result. The Pesticide Recommendation module recommends appropriate pesticides based on the identified pest type, considering factors, thus aiding users in making informed decisions. The User Interface module offers an intuitive interface for users to interact with the system, providing functionalities for image upload, and displaying classification results and pesticide recommendations in a user-friendly format.

The Deployment module handles the deployment of the system on web infrastructure and integration with web browser, reliability, and security. The The Testing and Validation module conducts testing and validation to ensure system functionality, accuracy, and reliability, thus ensuring the effectiveness of the system in addressing agricultural pest management challenges. Regular updates and retraining ensure that the model can react to new pest kinds and emerging threats and updates to ensure continued efficiency and effectiveness of the system in assisting users.

CHAPTER 5

RESULTS AND DISCUSSIONS

5.1 OUTPUT

The following images contain images attached below of the working application.

Insect Pest Classification and Pesticide Recommendation Webpage

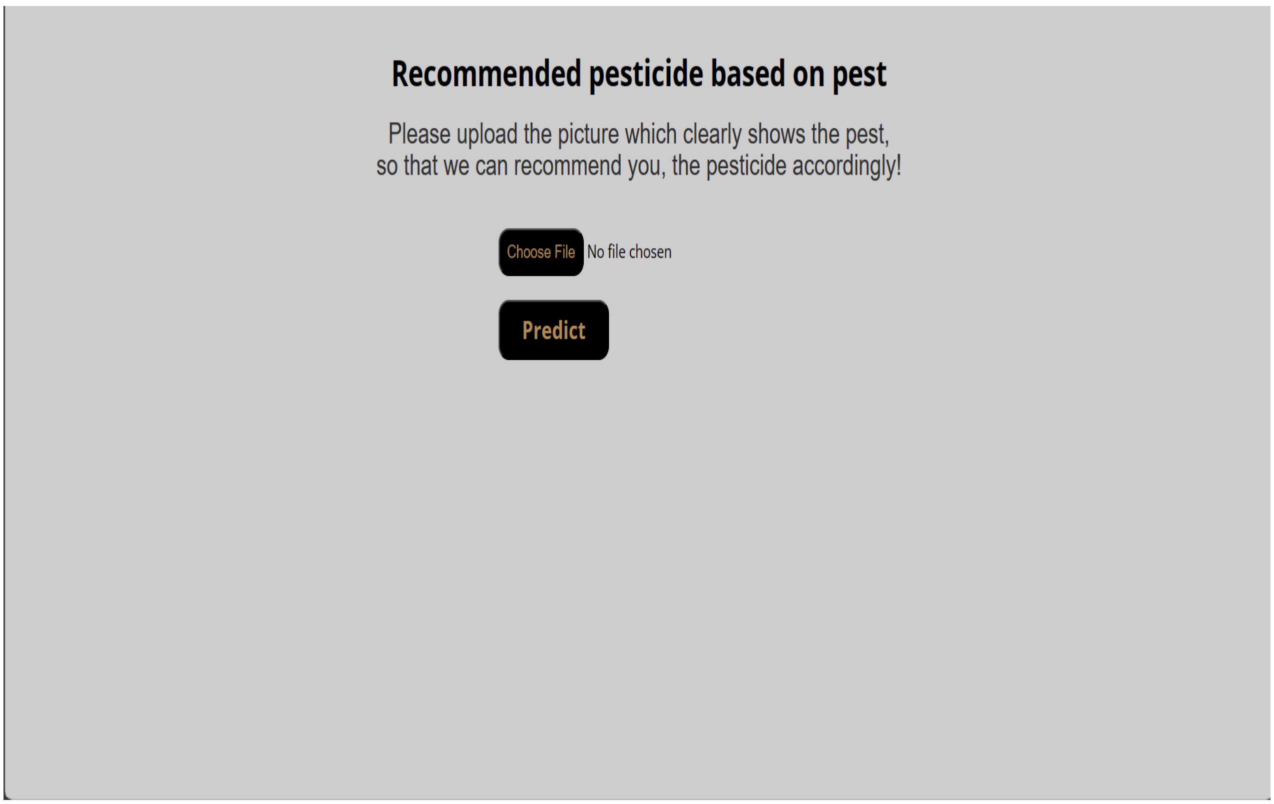

Fig 5.1.1: Home page for the User to upload the insect pest image

Uploading Insect pest image

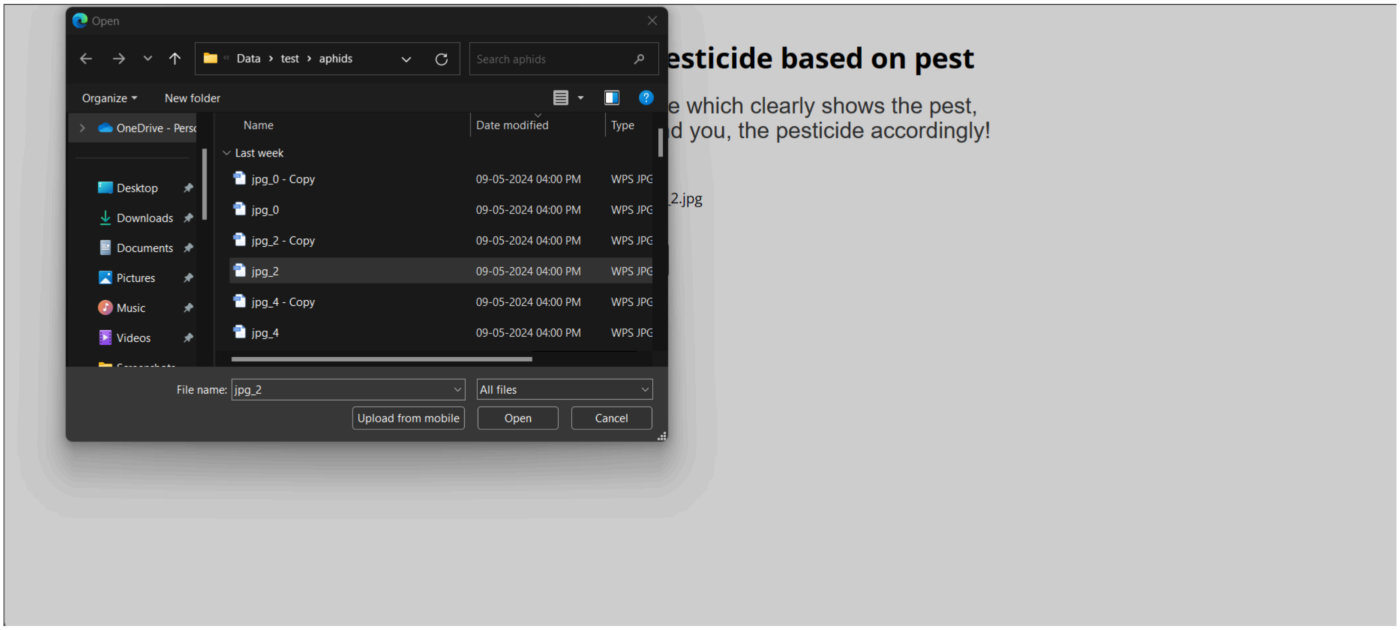

Fig 5.1.2: Uploading Insect pest image for classify and recommend suitable pesticide

Output of Insect pest classification and recommended pesticides

Fig 5.1.3: Recommendation of suitable pesticides based on the identified pest

Training and Testing Accuracy Graph:

The proposed model is evaluated and the testing and training accuracy graph is obtained. Spliting the dataset into training and validation sets (e.g., 80-20 split). Training the model using the training set, adjusting hyperparameters to optimize performance. Employ techniques such as dropout and batch normalization to prevent overfitting.The training and testing accuracy rate of the model is attached in the below figure 5.1.4

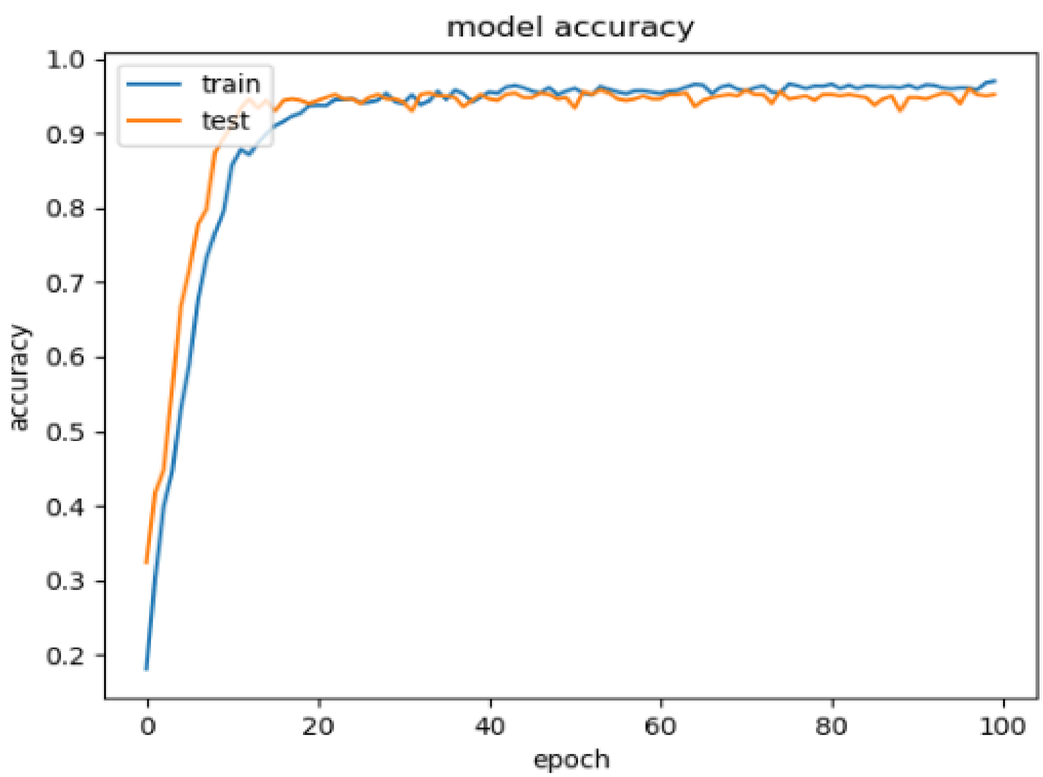

Fig 5.1.3: Training and Testing Accuracy Graph

Training and Testing Loss Graph:

The proposed model is evaluated and the testing and training loss graph is obtained.The training loss typically starts high and gradually decreases as the model learns the underlying patterns in the training data. An ideal graph will show a steady decline in training loss, indicating effective learning.The training and testing loss rate of the model is attached in the below figure 5.1.5

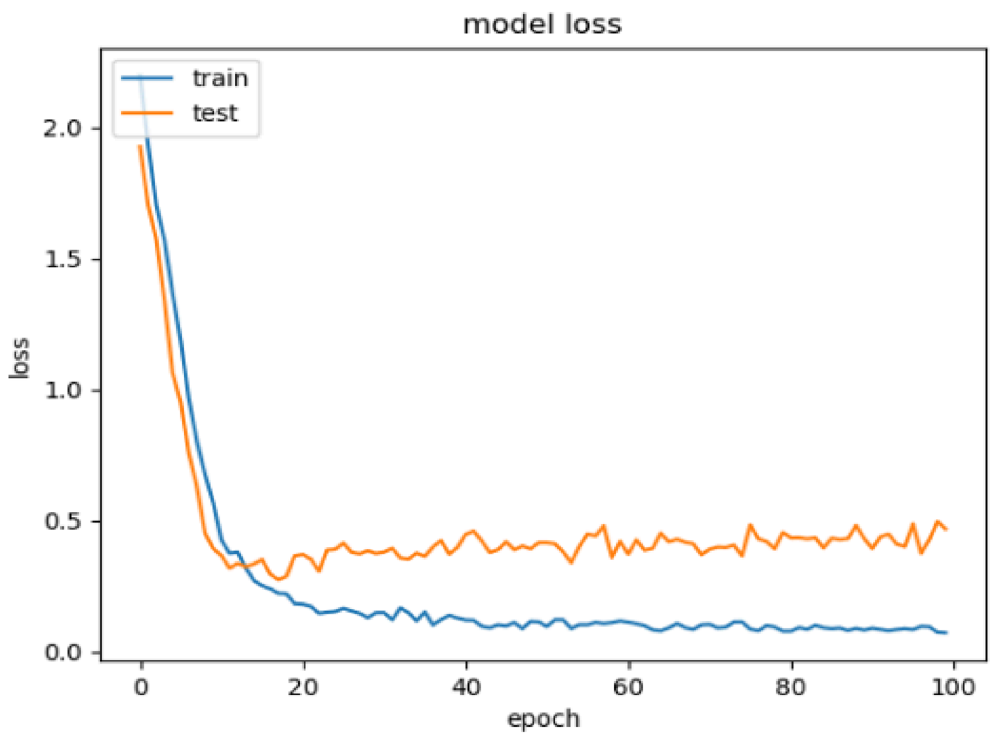

Fig 5.1.4: Training and Testing Loss Graph

5.2 RESULT

The outcomes of the project implementation and deployment of the Insect Pest Classification and Recommendation of Pesticides using Random Forest Algorithm. Through rigorous testing the is validated to find the capability to classify insect pests and the system has achieved a accuracy of 96%. The system's performance metrics, includes classification accuracy and its reliability and effectiveness in identifying pests and providing relevant recommendations. Users interact with an intuitive user interface, enabling seamless navigation and access to functionalities such as image upload, manual data input, and viewing classification results and pesticide recommendations. The efficiency of the system can be further increased by training the model iteratively but it may also increase the training time of the model.Overall, the project's results demonstrate a significant advancement in leveraging machine learning and technology to address agricultural pest management challenges, offering a practical and efficient solution for enhancing agricultural productivity and sustainability while minimizing environmental impact.

CHAPTER 6

CONCLUSION AND FUTURE ENHANCEMENT

6.1 CONCLUSION

In conclusion, the insect pest classification and pesticide recommendation system, uses Convolutional Neural Networks (CNNs) and Random Forest algorithms, presents a solution for agricultural pest management. By harnessing machine learning techniques, this system accurately identifies insect pests and provides suitable pesticide recommendations, improving crop yield and quality. The system provides a effective way to solve the challenges posed by insect pests in agriculture. Moreover, its user-friendly interface ensures accessibility to farmers helping them to make right decisions and optimize pesticide usage. This project signifies a step forward in providing farmers with a tools for efficient pest management.

6.2 FUTURE ENHANCEMENT

Expansion of Dataset: Continuously augmenting the dataset with additional insect pest images and metadata from diverse geographic regions and crop types can improve the system's accuracy and generalization capabilities.

Real-time Monitoring: Implementing real-time monitoring capabilities to track insect pest populations and environmental conditions would enable proactive pest management strategies and timely interventions.

APPENDIX

SOURCE CODE:

app.py:

```
from flask import Flask, render_template, request, Markup
import pandas as pd
from utils.fertilizer import fertilizer_dict
import os
import numpy as np
from keras.preprocessing import image
from keras.models import load_model
import pickle

classifier = load_model('Trained_model.h5')
classifier._make_predict_function()

app = Flask(__name__)

def pred_pest(pest):
    try:
        test_image = image.load_img(pest, target_size=(64, 64))
        test_image = image.img_to_array(test_image)
        test_image = np.expand_dims(test_image, axis=0)
        result = classifier.predict_classes(test_image)
        return result
    except:
        return 'x'
```

```
@app.route("/")
@app.route("/PesticideRecommendation.html")
def pesticide():
    return render_template("PesticideRecommendation.html")

@app.route("/predict", methods=['GET', 'POST'])
def predict():
    if request.method == 'POST':
        file = request.files['image']  # fetch input
        filename = file.filename

        file_path = os.path.join('static/user uploaded', filename)
        file.save(file_path)

        pred = pred_pest(pest=file_path)
        if pred == 'x':
            return render_template('unaptfile.html')
        if pred[0] == 0:
            pest_identified = 'aphids'
        elif pred[0] == 1:
            pest_identified = 'armyworm'
        elif pred[0] == 2:
            pest_identified = 'beetle'
        elif pred[0] == 3:
            pest_identified = 'bollworm'
        elif pred[0] == 4:
            pest_identified = 'earthworm'
        elif pred[0] == 5:
```

```
            pest_identified = 'grasshopper'
        elif pred[0] == 6:
            pest_identified = 'mites'
        elif pred[0] == 7:
            pest_identified = 'mosquito'
        elif pred[0] == 8:
            pest_identified = 'sawfly'
        elif pred[0] == 9:
            pest_identified = 'stem borer'

        return render_template(pest_identified + ".html",pred=pest_identified)
```

model.py:

```
from keras.models import Sequential
from keras.layers import Convolution2D
from keras.layers import MaxPooling2D
from keras.layers import Flatten
from keras.layers import Dense, Dropout
from keras import optimizers
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import accuracy_score, classification_report
from sklearn.model_selection import train_test_split



classifier = Sequential()


classifier.add(Convolution2D(32, 3,  3, input_shape = (64, 64, 3), activation = 'relu'))
```

```
classifier.add(MaxPooling2D(pool_size =(2,2)))

classifier.add(Convolution2D(32, 3,  3, activation = 'relu'))
classifier.add(MaxPooling2D(pool_size =(2,2)))

classifier.add(Convolution2D(64, 3,  3, activation = 'relu'))
classifier.add(MaxPooling2D(pool_size =(2,2)))

classifier.add(Flatten())

classifier.add(Dense(256, activation = 'relu'))
classifier.add(Dropout(0.5))
classifier.add(Dense(10, activation = 'softmax'))

classifier.compile(
        optimizer = 'adam',
        loss = 'categorical_crossentropy',
        metrics = ['accuracy'])

from keras.preprocessing.image import ImageDataGenerator
train_datagen = ImageDataGenerator(
    rescale=1./255,
    shear_range=0.2,
    zoom_range=0.2,
    horizontal_flip=True)

test_datagen = ImageDataGenerator(rescale=1./255)
```

```
training_set = train_datagen.flow_from_directory(
        'Data/train',
        target_size=(64, 64),
        batch_size=32,
        class_mode='categorical')

test_set = test_datagen.flow_from_directory(
        'Data/test',
        target_size=(64, 64),
        batch_size=32,
        class_mode='categorical')

model = classifier.fit_generator(
        training_set,
        steps_per_epoch=100,
        epochs=100,
        validation_data = test_set,
        validation_steps = 6500
      )

X_train, X_test, y_train, y_test = train_test_split(features, y, test_size=0.2, random_state=42)

rf_classifier = RandomForestClassifier(n_estimators=100, random_state=42)

rf_classifier.fit(X_train, y_train)
```

```
predictions = rf_classifier.predict(X_test)

import h5py
classifier.save('Trained_Model.h5')

print(model.history.keys())
import matplotlib.pyplot as plt

plt.plot(model.history['acc'])
plt.plot(model.history['val_acc'])
plt.title('model accuracy')
plt.ylabel('accuracy')
plt.xlabel('epoch')
plt.legend(['train', 'test'], loc='upper left')
plt.show()

plt.plot(model.history['loss'])
plt.plot(model.history['val_loss'])
plt.title('model loss')
plt.ylabel('loss')
plt.xlabel('epoch')
plt.legend(['train', 'test'], loc='upper left')
plt.show()
```

PesticideRecommendaation.html:

```
<!DOCTYPE html>
<html lang="en">
```

```
<head>
  <meta charset="utf-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <title>Irrigreat</title>

   <link href='http://fonts.googleapis.com/css?family=Open+Sans:400,300,400italic,600,700'
rel='stylesheet'
    type='text/css'>
   <link rel="preconnect" href="https://fonts.gstatic.com">
                                                                                            <link
href="https://fonts.googleapis.com/css2?family=Recursive:wght@500&display=swap"
rel="stylesheet">
         <link       href='http://fonts.googleapis.com/css?family=Damion'       rel='stylesheet'
type='text/css'>
  <link href="{{ url_for('static', filename='css/bootstrap.min.css') }}" rel="stylesheet">
  <link href="{{ url_for('static', filename='css/font-awesome.min.css') }}" rel="stylesheet">
  <link href="{{ url_for('static', filename='css/templatemo-style.css') }}" rel="stylesheet">
     <link   rel="shortcut   icon"   href="{{   url_for('static',   filename='img/favicon.ico')   }}"
type="image/x-icon" />
  <script type="text/JavaScript" src="{{ url_for('static', filename='js/cities.js') }}"></script>
  <style>
    html body {
      background-color: #e4e4e4;
  }
  </style>
</head>
```

```
<body>
{% block body %}
<br /><br />
<h2 style="text-align: center; margin: 0px; color: black">
  <b>Recommended pesticide based on pest</b>
</h2>
<h3 align="center">Please upload the picture which clearly shows the pest,<br>
            so that we can recommend you, the pesticide accordingly!</h3>
<br />

<div
  style="
    width: 350px;
    height: 40rem;
    margin: 0px auto;
    color: black;
    border-radius: 25px;
    padding: 10px 10px;
  "
>
    <section>
                          <form action="/predict" method="post" enctype="multipart/form-data"
onsubmit="showloading()">
                <input type="file" name="image" class="upload" style="font-size: 15px;">
                <br>
                <input type="submit" value="Predict" style="color: #c79c60; font-weight: bold;
                background-color: black;
                width: 130px; height:50px; border-radius:12px; font-size: 21px;">
```

```
            </form>
    </section>
</div>
</body>
</html>
```

REFERENCES

[1] "**Large-Scale Insect Pest Image Classification**" by Thanh-Nghi Doan (published in **2023**).

[2] "**A Large-Scale Benchmark Dataset for Insect Pest Recognition**" by Rajak (Published in **2022**)

[3] "**Position dependent identification for field moth images using deep learning**" by Chenglu Wen in **2022.**

[4] "**Pest detection and recognition using k-means clustering algorithm**" by Phil Birch, Rupert C D (Published in **2021**).

[5] "**A study on various data mining techniques for crop yield prediction**" by Gandge, Yogesh, and Sandhya (Published in **2021**)

[6] "**Pest identification in images using SVM classifier**" by Amsini Ram(Published in **2021**)

[7] "**Classification of Medicinal Plants Leaves Using Deep Learning Technique**" by H. Chanyal, R.K. Yadav, and D.K.J. Sain (Published in **2020**)

[8] "**Crop Recommendation System to Maximize Crop Yield using Machine Learning Technique** "by Rajak, Rohit Kumar (Published in **2020**)

[9] "**Herbal Leaves Classification Based on Leaf Image Using CNN Architecture Model VGG16**,**202**" by UN Oktaviana, GW Wicaksono (Published in **2020**)

[10] "**A Review on Extreme Learning Machine**" by J. Wang, S. Lu, S.H. Wang, and Y.D. Zhang(Published in **2019**)

[11] "**Iris Features Extraction and Recognition Based on the Scale Invariant Feature Transform (SIFT)**" by M.A. Taha, H.M. Ahmed, and S.O. Husain (Published in **2019**)

[12] "**Crop Recommendation System for Precision Agriculture**" by Dighe, Deepti (Published in **2019**)

[13] "**A Recommendation System for Farmers**" by Mokarrama, Miftahul Jannat, and Mohammad Shamsul Arefi (Published in **2019**)

[14] "**A study on various data mining techniques for crop yield prediction**" by Gandge, Yogesh, and Sandhya (Published in **2018**)

[15] "**Plant Disease and Pest Detection Using Deep Learning-Based Features** " by M. Türkoğlu and D. Hanbaya (Published in **2018**)