NABTA

July 2022

Acknowledgement

At First, at last, and all the time, for everything in my life. Nothing could be done without God's permission, and no success could be gained without his mercy. Thanks to **Dr. Esraa M. Elhariri** for her very much support and encouragement to accomplish this project in a professional and valuable way. She provided us with invaluable advice and helped us in difficult periods, her motivation and help contributed tremendously to the successful completion of the project. Also, thanks to **Dr. Mostafa Rabea** for his valuable help and thanks to **Dr. Asmaa Hashem** for her very much support.

Sincere thanks.

Abstract

The agriculture sectors which includes crop and livestock production as well as fisheries face many risks, such as climate and market volatility, pests and diseases, an ever-increasing number of protracted crises and conflicts, and extreme weather events resulting in loss of nutritional value, over-ripen and changes at texture, appearance, and flavor, and this affects the quality of the agricultural crops. This leads to losses of more than 40 billion dollars in only Africa. Nabta project aims to develop an automated helpful system for agriculture that can be useful for farmers, farming lovers, and agricultural companies. It can perform the main agricultural tasks accurately by using deep learning to solve the problems faced by farmers, and farming lovers such as crop identification, disease recognition, ripeness assessment, and help with how to care about a specific plant.

Nabta helps in the early detection of the disease and yields assessment to reduce loss in crops. NABTA project achieved an accuracy of 99% for plant recogination and an average accuracy of 99% for diseases recogination of eight crops. Moreover, NABTA project achieved mean average precision (mAP) of 91% for strawberry ripeness classification.

Contents

List of Figures

Chapter 1

Introduction

1.1 Overview

For decades, agriculture has been associated with the production of essential food crops. Nowadays, agriculture above and beyond farming includes forestry, dairy, fruit cultivation, poultry, beekeeping, mushroom, arbitrary, etc. Today, processing, marketing, and distribution of crops and livestock products, etc. Are all acknowledged as part of current agriculture? Thus, agriculture could be referred to as the production, processing, promotion, and distribution of agricultural products. Agriculture plays a critical role in the entire life of a given economy.

Agriculture is the backbone of the economic system of every country. In addition to providing food and raw material, agriculture provides employment opportunities to a very high percentage of the population.

In most parts of the world, agriculture is an essential source of livelihood, which entails hard work, but it contributes to the nation's food safety and health. Agriculture was the primary source of the economy before the industrial revolution. With the many coming up trade options, many people depend on agriculture as the main source of income. Agriculture is the most peaceful and environmentally friendly method. It is a very reliable source of life for humanity, as well as one of the honest sources of income. Many people from developing countries rely for their livelihood on agriculture.

One of the prime factors in ensuring consistent marketing of crops is product quality. For many crops, the main indicator of product quality from the customer's perspective is crop ripeness and health. Also, one of the most worrying issues for producers is the product's appearance as it has a high influence on the product's quality and consumer preferences. Fruit ripening is characterized by the development of color, flavor, texture, and aroma. The actual time from anthesis until full maturity can vary tremendously among species/cultivars due to genetic and environmental differences. Even between fruit on the same plant, fruit development, and ripening can take more or less time depending on local microclimate conditions and differences in sink/source relations within the plant.

In addition, when a fruit is harvested, the time of anthesis of a particular fruit is generally unknown, as is its full history. Monitoring and controlling fruit and vegetable ripeness has become a very important issue in the crops industry since ripeness is perceived by customers as the main quality indicator. Also, the product's appearance is one of the most worrying issues for producers as it has a highl influence on the product's quality and consumer preferences. However, up to this day, optimal harvest dates and prediction of storage life are still mainly based on subjective interpretation and practical experience.

Plant pests and diseases affect crops, causing significant losses to farmers and threatening food security. The spread of plant pests and diseases has increased dramatically in recent years. Globalization, trade, and climate change, as well as reduced resilience in production systems due to decades of agricultural intensification, have all played a part. Plant pests and diseases can quickly spread to several countries and reach epidemic proportions. Outbreaks and upsurges can cause huge losses to crops and pastures, threatening the livelihoods of vulnerable farmers and the food and nutrition security of millions at a time [1].

(a) Apple (b) Apple (c) Grape (d) Corn

Figure 1.1: Plant pests and diseases

Plant pests and diseases spread in three principal ways:

- trade or other human - migrated movement.
- environmental forces - weather and wind borne.
- insect or other vector-borne - pathogens.

Hence, automation of this process is a big gain at agriculture and industry fields. For agriculture, it may be used to develop automatic harvest systems and saving crops from damages caused by environmental changes. On the other hand, for industry, it is used to develop automatic sorting system or checking the quality of fruits to increase customer satisfaction level [2]. So, an objective and accurate ripeness and health assessment of agricultural crops is important in ensuring optimum yield of high quality products. Moreover, identifying physiological and harvest maturity of agricultural crops correctly, will ensure timely harvest to avoid cutting of either under- and over-ripe agricultural crops [3, 4].

Agricultural sustainability can be attained through vision enabled autonomous machines that work together as a phenomenon to ensure global food security. The demand for efficient as well as reliable food production techniques is rapidly increasing day by day. Computer vision tagged with machine learning approaches grabbed considerable attention for research to meet this demand through analyzing and understanding the input images from humans, robots, drones, sensors, satellites, etc.
Began to use computer vision and machine learning as well as deep learning techniques for attaining increased agricultural productions. Additionally, with the help of the above-mentioned techniques different agricultural activities, such as crop health monitoring, weed, disease, pest detection, etc. have also been reviewed to overcome the current challenges and explore the future opportunities for smart farming with low cost and high efficiency [1].

1.2 Problem Definition

Agricultural crops at greenhouse, fields and storage containers are affected by changes at climate such as temperature and humidity levels, etc. These changes can cause diseases, loss of nutritional value, over-ripen and changes at texture, appearance and flavor and this affect the quality of the agricultural crops. Crop diseases are a major threat to food security, but their rapid identification remains difficult in many parts of the world due to the lack of the necessary infrastructure. Monitor the ripeness process and health of crops is very important issue at agricultural and industry. The combination of increasing global smartphone penetration and recent advances in computer vision made possible by deep learning has paved the way for smartphone-assisted disease diagnosis.

1.3 Project Objectives

The main objectives of this project to be accomplished within a timeline and with available resources are:

1. Developing an automated helpful system for agriculture that can be useful for farmers, farming lovers, and agricultural companies. It can perform the main agricultural tasks accurately using deep learning to solve the problems faced by farmers such as:
   (a) Crop identification.
   (b) Identification of diseases.
   (c) Detection of insects.
   (d) Fruit detection.
   (e) Ripeness Assessment.
2. Early detection of the disease helps farmers to make the right decision about treatment and preventive things such as
   (a) Fertilizers.
   (b) Herbicides.
   (c) Insecticides.

1.4 Project Plan

- Building a theoretical concepts background including (Literature review, Survey on similar systems and project proposal).
- Choosing specific crops and preparing images dataset.
- Studying chosen crops nature and what are the most important features for ripeness assessment and disease recognition?

- Studying different deep learning models for classification.

    1. Pre-processing and features extraction.
    2. Classifications and decision making
    3. Experimental results and evaluation

1.5 Organization

We structure the rest of this project as follows: Chapter (2) presents a survey of some work in this area and related problems in details, Chapter (3) is an overview of the methods used in the proposed system, Chapter (4) gives a detailed description of the phases of the proposed classification system, Experimental results and comparative analysis are discussed in chapter (5), Challenges and future trends are presented in chapter (6).

Chapter 2

Literature Review

2.1 Introduction

Recently, utilizing computer vision and machine learning (ML) in food products has become very wide spread, especially for products where measuring color or other spectral features enables estimating the ripeness and health stage. Images acquisition is performed using a physical image sensor, then a dedicated computing hardware and software are used for the purpose of images analysis with the objective of performing a predefined visual task. Computer vision techniques allow to evaluate fruits maturity and health accurately without destruction because image processing is a visual technology to observe and analyze an object without touching the object being observed. As discussed at the previous chapter, this project presents:

1. A deep learning-based plant recognition system to identify plant name.

2. a deep learning-based plant diseases recognition system to identify plant disease.

3. YOLO-based object detection system for monitoring the ripeness process of plants via investigating and classifying the different maturity/ripeness stages.

The proposed systems based on learned features. In recent years, there has been substantial work in the computer vision and ML fields which tackling the ripeness and health assessment and classification problem of fruits/vegetables. This chapter reviews a survey about current approaches which tackling the ripeness and health assessment and classification problem of fruits/vegetables.

2.2 Identification of Plants using Deep learning

B. K. Varghese et. al proposed an android application called INFOPLANT to identify plants using the CNN (Convolutional Neural Network). Transfer learned MobileNet model is used in this system. The model is trained with customized dataset which is created from video of plant species. The dataset is then converted and stored as .tflite file. The application predicts the input plant image with the tflite model. After prediction the application will check all the labels and find the label with maximum probability which gives the plant name as the output. The obtained output is then connected to firebase. The output will be the details like biological name, common name, location, nutrient requirement needed by the plant and the medicinal value of the plant. The proposed model achieved a prediction result with accuracy of 99% and validation accuracy of 95% [5].

S. A. Riaz et. al presented a plant identification method using multipath multi-deep convolutional network. The proposed architecture illustrated consists of multiple CNN blocks, max pooling layers, flatten layer and soft-max layer for the classification of the input plant images. Each block consists of three convolution, one batch normalization, one max-pooling layer and one dense layer. The features extracted from one block is concatenated with the features from second block. The soft-max layer classifies the plant species. This study used two datasets LeafSnap and MalayaKew and achieved accuracy of 99.38% and 99.22%, respectively [6].

Kaya et. al suggested the concept of transfer learning for plants classification focused on Deep Learning. This indicates the impact of four separate transfer training models on plant classification for four available databases. Finally, their theoretical research reveals that transfer learning offers a basis of plant classification self-estimating and analysing. They use certain common formats including End-to-End, Fine modulation, Fine modulation Cross Dataset, Deep Integrated Finetuning, Classification by RNN-CNN. Using total number of 54,306 images, the proposed approach achieved accuracy of 98.70% [7].

N. Manasa et. al proposed a plant identification method using watershed algorithm and convolutional neural network. The input image undergoes various preprocessing stages when the leaf is surrounded by multiple leaves watershed algorithm is used to separate each leaf. The proposed method consists of two phases training and testing phase. In this model a pre-trained convolutional neural network is used to solve classification problem. Stochastic Gradient Descent algorithm with momentum is used to train the network and is successfully trained with a validation accuracy of 100% [8].

Beikmohammadi, and Faez. proposed a method that uses transfer learning to identify plant leaf for classification. This method uses the MobileNet model as a feature extractor . With the help of image label and the feature vector logistic regression classifier was trained to predict the target class. The presented method works directly with RGB images which eliminate the use of pre-processing stage and hand-crafted feature extraction. The proposed method is evaluated on flavia dataset and LeafSnap dataset and achieved accuracy of 99.6% and 90.54% respectively [9].

2.3 Leaf Diseases Recognition using Deep Learning

Peng Wang et. al proposed a deep learning model, which is called the Coordination Attention EfficientNet (CA-ENet) to identify different apple diseases. First, a coordinate attention block is integrated into the EfficientNet-B4 network, which embeds the feature's spatial location information by channel attention to ensure that the model can learn both the channel and spatial location information of important features. Then, a depth-wise separable convolution is applied to the convolution module to reduce the number of parameters, and the h-swish activation function is introduced to achieve the fast and easy-to-quantify process. Afterward, 5,170 images are collected in the field environment at the apple planting base of the Northwest A&F University, while 3,000 images are acquired from the PlantVillage public data set. Also, image augmentation techniques are used to generate an Apple Leaf Disease Identification Data set (ALDID), which contains 81,700 images. The experimental results show that the accuracy of the CA-ENet is 98.92% on the ALDID, and the average F1-score reaches 0.988, which is better than those of common models such as the ResNet-152, DenseNet-264, and ResNeXt-101 [10].

HELONG YU et. al presented a method based on deep learning to identify various corn illnesses. A coordinate attention block is first integrated into the

EfficientNet-B4 network. This block embeds the spatial location information of the feature by channel attention. The convolution module is then given a depth-wise separable convolution to reduce the number of parameters, and the h-swish activation function is added to produce the quick and simple quantification process. The findings of the experiment show that the method has the strongest identification effect on 32-means samples, and the diagnostic recall for leaf spot, rust, and grey spot illness is, respectively, 89.24%, 100%, and 90.95%. On 32-means samples, VGG-16 and ResNet18 also produce the greatest diagnostic outcomes, with average diagnostic accuracy of 84.42 percent and 83.75 percent, respectively. Additionally, on the 64-means samples, Inception v3 (83.05%) and VGG-19 (82.63%) perform the best. The method used has an average diagnostic accuracy of 93% for the three corn diseases. The corn data set used from the Crop Disease Recognition of the 2018 Artificial Intelligence Challenger Competition (challenger.ai) Three hundred images of each disease are chosen, resulting in a total of 900 disease images [11].

Guowei Wang et. al proposed an improved ResNet50 model for maize disease identification. The framework: PaddlePaddle adjusted the learning strategy through the inclined triangle learning rate, increases L2 regularization to reduce overfitting, and adopts an exit strategy and ReLU incentive function. Then, the first convolution kernel of the ResNet50 model was modified into three $3x3$. the data enhancement methods, such as brightening, translation, and flipping, were used to expand the data set. The image recognition accuracy in the dataset is 98.52%, the image recognition accuracy in the farmland is 97.826%, and the average recognition speed is $204ms$, which meets the accuracy and speed requirements of the maize field spraying operation[12].

Miaomiao Ji et. al proposed a united convolutional neural networks (CNNs) architecture based on an integrated method. The proposed CNNs architecture was designed to distinguish leaves with common grape diseases i.e., black rot, esca, and isariopsis leaf spot from healthy leaves. The proposed the UnitedModel which is a united CNNs architecture based on InceptionV3 and ResNet50 and can be used to classify grape images, the combination of multiple CNNs enables the proposed UnitedModel to extract complementary discriminative features. Thus the representative ability of UnitedModel has been enhanced. The UnitedModel has been evaluated on the hold-out PlantVillage dataset and has been compared with several state-of-the-art CNN models. The experimental results have shown that UnitedModel achieved the best performance on various evaluation metrics. The UnitedModel achieved an average validation accuracy of 99.17% and a test accuracy of 98.57%, which can serve as a decision support tool to help farmers identify grape diseases [13].

2.4 Ripeness Stages Identification using Deep Learning

Rucha Thakur et. al suggested an automated system that predict the ripeness level of strawberry fruit, using a convolutional neural network. The surface color of strawberry fruit determines its ripeness level. CNN is used to extract color, size and shape features from strawberry surfaces.In the system, the input image is fed to the Convolutional neural network. In CNN, the image is segmented for easy analysis. Feature Extraction provided by CNN to automatically extract all the necessary features through the pool of Network. Based on the Feature Extraction, training is done. The testing data is passed for two stage classification based on features: size, shape, and surface color. The stage 1 classification is done between the raw and ripe, if the output is ripe it is passed to the stage 2 classification, else the output is Raw. In stage 2 classification is done between ripe and damage. The output of stage 2 determines whether strawberry is ripe or damaged. The classification output along with the image classified is shown on the GUI.The dataset of the system includes images of three classes of Strawberry fruit: premature, mature and over mature. Since the images of premature and over-mature strawberries are acquired, the dataset for premature and over-mature strawberries are created anually with the help of a standard camera and downloaded through web images. The images for mature strawberries are acquired through google images. The dataset includes a total of 240 images for training purpose and 60 images for the testing purpose. The developed system achieves 91.6% accuracy [14].

Yuanyuan Shao et. al presented a method for assessment of strawberry ripeness using Hyperspectral Imaging. Portable hyperspectral imaging was used for field and indoor spectra acquisition of the strawberries to extract the effective wavelengths. Two classifiers, partial least squares - discriminant analysis (PLS-DA) and least squares - support vector machine (LS-SVM) were used for ripeness assessment. The results showed that the overall accuracy of all classifiers for field assessment ranged from 91.7% to 96.7%, slightly lower than for indoor assessment. Furthermore, the LS-SVM model combined with effective wavelengths with the CARS method performed better for the field assessment of strawberry ripeness, providing an accuracy of 96.7%. It can be concluded that hyperspectral imaging can be used for the real-time assessment of strawberry ripeness in the field. Finally, the performance of PLS-DA and LS-SVM models was assessed with discrimination accuracy for strawberries of each ripeness level. data set Field experiments were conducted in the strawberry garden at New Green Vegetable Cooperative in Tai'an City, Shandong Province, China on a sunny morning in March 2019 [15].

2.5 Summarizes the Presented Exhaustive

Table (2.1) summarizes the presented exhaustive survey of state-of-the art studies related to plant recognition, diseases recognition, and ripeness assessment approaches based on deep Learning.

Table 2.1: Summarizes the presented exhaustive.

Author	Objectives	Deep Learning Model	Dataset	Performance Measures
2020 [5]	Identification of plants using the CNN (Convolutional Neural Network).	MobileNet	Dataset which is created from video of plant species. The dataset is then converted and stored as .tflite file.	Prediction accuracy = 99%, Validation accuracy = 95%.
2020 [6]	Identification of plants using multipath multi deep convolutional network.	MPF-CNN	Leafsnap and MalayaKew datasets.	LeafSnap dataset: Accuracy= 99.38% MalayaKew dataset: Accuracy = 99.22%.
2019 [7]	Identification of plants using Transfer learning method use on deep learning.	DNN	Using total number of 54,306 images.	Accuracy = 98.70%.
2019 [8]	Identification of plants using watershed algorithm and convolutional neural network.	Multi-scale fusion convolutional neural network (MSF-CNN) for plant leaf recognition at multiple scales.	Leafsnap and MalayaKew datasets.	Validation accuracy = 100%.

Continued on next page

Table 2.1: Summarizes the presented exhaustive. (Continued)

Author	Objectives	Deep Learning Model	Dataset	Performance Measures
2018 [9]	Identification of plant leaf for classification using transfer learning.	MobileNet	Flavia and LeafSnap datasets.	Flavia dataset: Accuracy = 99.6%. LeafSnap dataset: Accuracy = 90.54%.
2021 [10]	Identification of different apple diseases.	EfficientNet (CA-ENet).	A total of 5,170 images are collected in the field environment at the apple planting base of the Northwest A&F University, while 3,000 images are acquired from the PlantVillage public data set.	Accuracy = 98.92% on the Apple Leaf Disease Identification F1-score = 98.8%.
2021 [11]	Addressing the problem of accurate diagnosis of corn crop diseases and the stages of corn growth and production.	Hybrid method based on K-means clustering and an improved deep learning model.	The corn data set used in this study is from the Crop Disease Recognition of the 2018 Artificial Intelligence Challenger Competition (challenger.ai).total of 900 disease images.	Accuracy = 98%.
2021 [12]	Improving the accuracy of maize disease identification.	ResNet50	The training and verification dataset from the PlantVillage dataset. The test data are collected in the actual environment.	Accuracy = 97.826%.
2020 [13]	Identification of leaves affected by common grape diseases.	ResNet50	PlantVillage dataset.	Validation accuracy = 99.17%. Test accuracy = 98.57%.

Continued on next page

Table 2.1: Summarizes the presented exhaustive. (Continued)

Author	Objectives	Deep Learning Model	Dataset	Performance Measures
2020 [14]	Predict the ripeness level of strawberry fruit.	CNN	The dataset includes a total of 240 images for training purpose and 60 images for the testing purpose from web images and google images.	Accuracy: 91.6%.
2021 [15]	Assessment of Strawberry Ripeness Using Hyperspectral Imaging.	Two classifiers, partial least squares - discriminant analysis (PLS-DA) and least squares - support vector machine (LS-SVM).	data set Field experiments were conducted in the strawberry garden at New Green Vegetable Cooperative.	Accuracy: 96.7%.

Chapter 3

Preliminaries

3.1 Introduction

This chapter presents a brief idea concerning the core concepts of smart farming system, which is able to recognize plant type, plant disease, ripeness stage and recommend, fertilizers, herbicides, and insecticides. Then it presents a brief idea about deep learning, and transfer learning.

3.2 Smart Agriculture System

Smart farming is a new concept that makes agriculture more efficient and effective using advanced information technologies. The latest advancements in connectivity, automation, and artificial intelligence enable farmers better to monitor all procedures and apply precise treatments determined by machines with superhuman accuracy. Farmers, data scientists, and engineers continue to work on techniques that allow for optimizing the human labor required in farming. With valuable information resources, improving daily, smart farming becomes a learning system and even more intelligent [16].

Plants are a pivotal part of our planet. Because of the existence of plants, the earth is called a green planet. They play a fundamental role in our life. To understand, we can't imagine our life without oxygen which plants provide, in addition to the relation of plants to our food, medicines, and furniture. Accordingly, we can say that plants were the basis of life. Plants are divided into smaller groups according to shared attributes. Plants recognition is complicated because plants are highly complex. The recognition process of familiar plants was easy for experts. Sometimes, especially in medicine, we need to identify prejudiced or toxic plants, botanists can do that facilely, but they must find a way to categorize the different species when there are millions of various plant species, which are composed of similar parts (roots, stems, leaves, etc.). Designing a plant recognition system is required to save time and decrease cost [17].

Early detection of plant disease is crucial to keep a crop healthy. The identification of plant disease can be a tedious task in many parts of the world due to the unavailability of the required equipment. Moreover, it has been reported that diseases and pests can reduce yields by more than 50%, and small farmers are the most vulnerable to this issue. For efficient crop management, the correct identification of the disease should be fast and cheap. With the huge leap in technological advancement in the past few years, high-resolution cameras, smartphones, and computers have become widespread and accessible to a large portion of the population [18]. All these factors can make an automated solution feasible, especially since deep learning techniques have shown their ability to perform exceptionally well on complex tasks. Early detection of plant diseases can help farmers effectively monitor the health of their culture in addition to make the best decision to avoid the spread of pathogens [19]. Figure (3.1) shows the general framework of Smart Agriculture System.

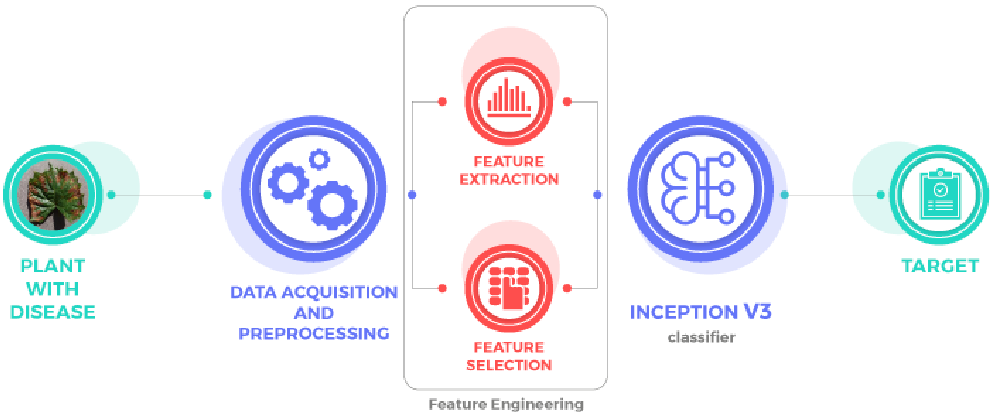

Figure 3.1: General framework of Smart Agriculture System

Moreover, Monitoring and controlling fruits and vegetables ripeness has become a very important issue in the crops industry, since ripeness is perceived by customers as the main quality indicator. Also, the product's appearance is one of the most worrying issues for producers as it has a high influence on product's quality and consumer preferences. However, up to this day, optimal harvest dates and prediction of storage life are still mainly based on subjective interpretation and practical experience. A standard smart farming system comprises three major components as depicted in Figure (3.1).

Data acquisition and preprocessing phase constitute a very important phase of any intelligent damage detection system and have a significant influence on the capability for damage detection and assessment. It is responsible for data collecting using various sensors and preprocessed the collected data for the feature engineering phase. Preprocessing includes data cleaning, data transformation, and dimension reduction.

Feature engineering phase is the key component of a successful ML algorithm, contributing to their performance. It aims to extract features that better represent the underlying problem from raw data and transforming them into a proper format for machine learning models.

Classification phase is responsible for classifying patterns into various categories using ML algorithms. These algorithms can be classified into three categories, namely, supervised learning, semi-supervised learning, and unsupervised learning based on the amount of required labeled data [20].

3.3 Data Preprocessing

3.3.1 Data augmentation

The high performance of deep CNN models is dependent on the availability of a large amount of training data. Unfortunately, a major issue generally faced by DL or ML is the scarcity of training data. As a result of insufficient training data, classification models suffer from the overfitting problem. To overcome the problem of overfitting, different regularization technologies are used, such as batch normalization and dropout layer usage. Another smart solution to this problem is data augmentation. It is the process of generating new samples that are similar to the training samples. However, unluckily, choosing inappropriate data augmentation methods is likely to lead to augment samples that are not sufficiently informative, which result in no impact or detrimental impact on the classifier's accuracy and robustness. Therefore, selecting the most appropriate data augmentation method according to the nature and requirements of the studied problem is key in achieving higher accuracy and robustness of classifiers, with a restricted number of generated training samples [20].

3.3.2 Feature Engineering

Feature Engineering is defined as the process of extracting features that better represent the underlying problem from raw data and transforming them into a suitable format for machine learning models, resulting in improving the performance of a trained model on unseen data. Feature engineering can be categorized into two categories, namely, feature extraction and feature selection. The main aim of feature extraction and selection is to

1. Reduce high-dimensional feature space to low-dimensional representation.
2. Focus on the most relevant data.
3. A void overfitting the data.
4. Improve the quality of feature space and hence the performance of machine learning algorithms such as learning time and accuracy [20].

3.3.3 Feature Extraction

Feature extraction is the process of extracting a set of new features from the original features through some mapping functions. Feature extraction methods can be practically classified into three groups: [20]

1. The hand-crafted features.
2. The learned ones.
3. hybrid features.

Hand-crafted features are those, which are extracted from an image following some certain hand-engineered predefined algorithms based on expert knowledge. There are a wide variety of state-of-the-art algorithms for the hand-crafted features. In contrast to the hand-crafted features, the **learned ones** are those set of features that are learned directly from raw input images by training a network with a labeled dataset to accomplish a certain task (e.g. face recognition). Convolutional Neural Networks (CNNs) are considered main examples of deep neural networks (DNNs), which can be used to extract learned features. The main idea behind the learned features approach is to discover data representations with multiple levels of abstraction to enable higher-level features of representing the semantics of the data, which provides better robustness to intra-class variability. Finally, hybrid features are those set of features that integrate both hand-crafted and learned features. In this chapter, we shall briefly discuss some of the popular learned features and hand-crafted features used in building SHM systems [20].

3.4 Deep Learning

In the last few years, the deep learning (DL) computing paradigm has been deemed the Gold Standard in the machine learning (ML) community. Moreover, it has gradually become the most widely used computational approach in the field of ML, thus achieving outstanding results on several complex cognitive tasks, matching or even beating those provided by human performance. One of the benefits of DL is the ability to learn massive amounts of data. The DL field has grown fast in the last few years and it has been extensively used to successfully address a wide range of traditional applications. More importantly, DL has outperformed well-known ML techniques in many domains, e.g., cybersecurity, natural language processing, bioinformatics, robotics and control, and medical information processing, among many others. Despite it has been contributed several works reviewing the State-of-the-Art on DL, all of them only tackled one aspect of the DL, which leads to an overall lack of knowledge about it [21].

3.4.1 Convolutional Neural Networks (CNNs)

Among the various deep neural network models, CNN is considered the most commonly used model for image classification. Standard CNN consists of several convolutional layers, pooling layers, and fully-connected (FC) layers. Figure (3.2) shows an example of CNN architecture. The main aim of the CNN is the automatic and adaptive learning of spatial hierarchies of useful features, from low-to high-level patterns [20]. Figure (3.2) shows an example of CNN architecture.

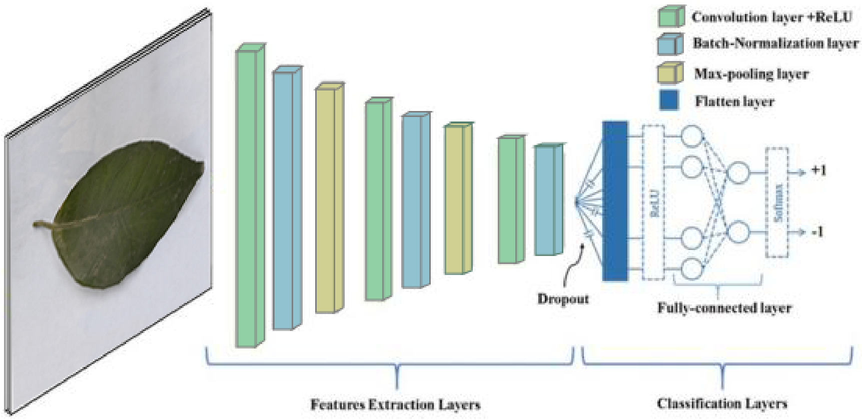

Figure 3.2: An example of CNN architecture.

1. **Convolutional Layer:**

   The convolutional layer is the key aspect of CNN. Given an input array A of size I a receptive field, and stride step N, it operates by applying three steps. The first step is an element-by-element multiplication between a sub-array of A, and a receptive field (both of size NxN). The second step is the summation of the multiplied values and adding bias to the summed values. The final step adds the final values to the output array. The receptive field is also often called the filter, or kernel. The weight values of a receptive field are initialized randomly [20].

2. **Pooling Layer:**

   Another key aspect of CNN is the down-sampling process performed by the pooling layer. It aims to achieve spatial invariance by reducing the resolution of the input feature map. Each pooled feature map corresponds to one feature map of the previous layer. Max and mean pooling are two types of pooling. The maximum values from an input array's sub-arrays are taken in max-pooling, while in mean pooling the mean values are taken. From the survey, max-pooling performance in image datasets outperformed mean pooling [20].

3. **Activation Layer:**

   The activation layer is a non-linear transformation function, widely used in the standard Artificial Neural Networks **(ANN)**. There are many different activation functions such as sigmoid, hyperbolic tangent ($tanh$), Rectified Linear Unit (**ReLU**), etc. It is applied after the convolution operation is completed to enable CNNs to avoid learning trivial linear combinations of inputs. All non-linear functions are restricted to output values except **ReLU**, which has only restricted outputs for its negative inputs. The features of **ReLU** make computation faster and more accurate. **ReLU** is computed according to equation (3.1) [20].

$$R(z) = max(0,z) \tag{3.1}$$

Where z is the input to a neuron.

4. **Softmax Layer:**

   For classifying input image, having a layer for prediction is necessary matter. This layer is responsible for classifying input images and is located at the end of CNN model. It can be any machine learning algorithms such as support vector machine **(SVM)**, multilayer perceptron **(MLP)**, etc. To date, using softmax function given by equation (3.2) is the most outstanding method [20].

   $$yk = \exp(\phi\ k) \,/\, (j\ \exp(\phi\ j)) \tag{3.2}$$

   Where ϕ is the neural network outputs and y is the probability of belonging to a class.

3.4.2 The Used CNN Deep Learning Model

There are various CNN models such as LeNet, AlexNet, ResNet, GoogleNet / InceptionNet, Depth based CNNs, VGG, Inception v2, Inception v3, Inception-ResNet, DenseNet, etc. Inception v3 deep learning model was selected because it is the advanced and optimized version of the inception V1 model. The Inception V3 model used several techniques for optimizing the network for better model adaptation [22].

- It has higher efficiency.
- It has a deeper network compared to the Inception V1 and V2 models, but its speed isn't compromised.
- It uses auxiliary Classifiers as regularizes [23].

3.4.3 Inception v3

Inception V3 by Google is the 3^{rd} version in a series of Deep Learning Convolutional Architectures [24]. It is mainly focuses on burning less computational power by modifying the previous Inception architectures and an image recognition model that has been shown to attain greater than 78.1% accuracy on the ImageNet dataset. The model is the culmination of many ideas developed by multiple researchers over the years. Figure (3.3) shows the structure of Inception V3 model.

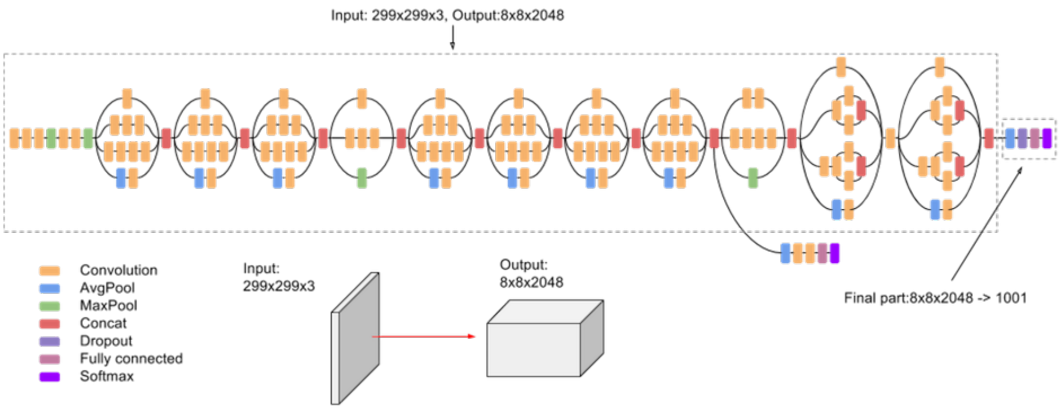

Figure 3.3: Structure of Inception V3 model.

The model itself is made up of symmetric and asymmetric building blocks, including convolutions, average pooling, max pooling, concatenations, dropouts, and fully connected layers. Batch normalization is used extensively throughout the model and applied to activation inputs. Loss is computed using Softmax [25].

3.4.4 Yolov5

YOLO is an acronym that stands for You Only Look Once. used Version 5, which was launched by Ultralytics in June 2020 and is now the most advanced object identification algorithm available due to its speed and accuracy. It is a novel convolutional neural network (CNN) that detects objects in real-time with great accuracy. This approach uses a single neural network to process the entire picture, then separates it into parts and predicts bounding boxes and probabilities for each component. These bounding boxes are weighted by the expected probability. The method "just looks once" at the image in the sense that it makes predictions after only one forward propagation run through the neural network. It then delivers detected items after non-max suppression (which ensures that the object detection algorithm only identifies each object once). Figure (3.4) shows scheme of the YOLOv5 architecture.

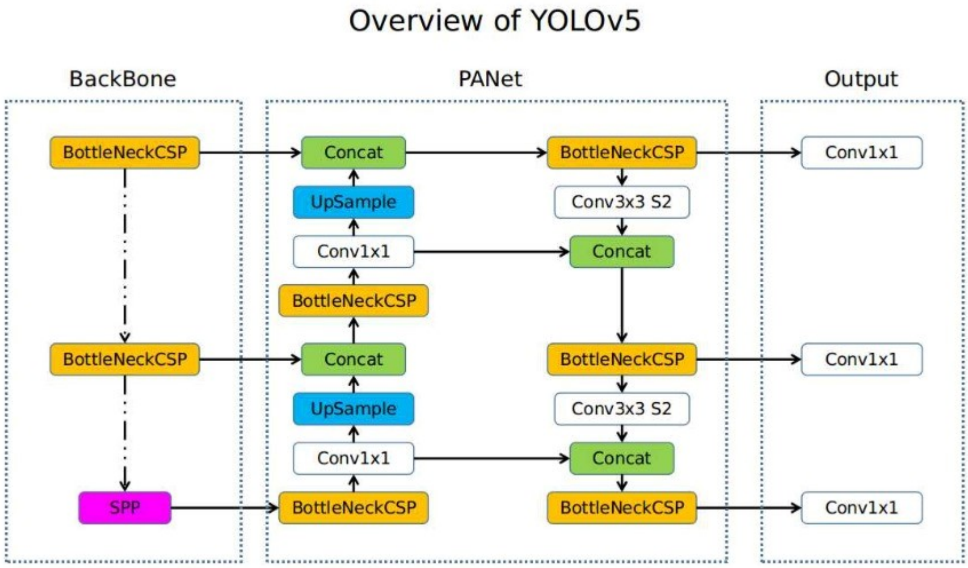

Figure 3.4: Scheme of the YOLOv5 Architecture.

The model itself is made up of symmetric and asymmetric building blocks, including convolutions, average pooling, max pooling, concatenations, dropouts, and fully connected layers. Batch normalization is used extensively throughout the model and applied to activation inputs. Loss is computed using Softmax [25].

1. **Backbone:** Model Backbone is mostly used to extract key features from an input image. CSP (Cross Stage Partial Networks) are used as a backbone in YOLO v5 to extract rich in useful characteristics from an input image.

2. **Neck:** The Model Neck is mostly used to create feature pyramids. Feature pyramids aid models in generalizing successfully when it comes to object scaling. It aids in the identification of the same object in various sizes and scales. Feature pyramids are quite beneficial in assisting models to perform effectively on previously unseen data. Other models, such as FPN, BiFPN, and PANet, use various sorts of feature pyramid approaches. PANet is used as a neck in YOLO v5 to get feature pyramids.

3. **Head:** The model Head is mostly responsible for the final detection step. It

uses anchor boxes to construct final output vectors with class probabilities, objectness scores, and bounding boxes [26].

3.5 Transfer Learning

Transfer learning is an emerging topic that may drive the success of machine learning in research and industry. The lack of data on specific tasks is one of the main reasons to use it, since collecting and labeling data can be very expensive and can take time, and recent concerns with privacy make difficult to use real data from users. The use of transfer learning helps to fast prototype new machine learning models using pre-trained models from a source task since training on millions of images can take time and requires expensive GPUs [27].

Step by step transfer learning process

1. **Obtain pre-trained model:**

   The first step is to choose the pre-trained model we would like to keep as the base of our training, depending on the task. Transfer learning requires a strong correlation between the knowledge of the pre-trained source model and the target task domain for them to be compatible.
   Here are some of the pre-trained models you can use: VGG-16, VGG-19, Inception V3, XCeption, ResNet-50, and etc.

2. **Create a base model:**

   The base model is one of the architectures such as ResNet or Xception which we have selected in the first step to be in close relation to our task. We can either download the network weights which saves the time of additional training of the model. Else, we will have to use the network architecture to train our model from scratch. There can be a case where the base model will have more neurons in the final output layer than we require in our use case. In such scenarios, we need to remove the final output layer and change it accordingly.

3. **Freeze layers:**

   Freezing the starting layers from the pre-trained model is essential to avoid the additional work of making the model learn the basic features. If we do not freeze the initial layers, we will lose all the learning that has already taken place. This will be no different from training the model from scratch and will be a loss of time, resources, etc.

4. **Add new trainable layers:**

   The only knowledge we are reusing from the base model is the feature extraction layers. And add additional layers on top of them to predict the specialized tasks of the model. These are generally the final output layers.

5. **Train the new layers:**

   The pre-trained model's final output will most likely differ from the output of model. For example, pre-trained models trained on the ImageNet dataset will output 1000 classes. However, need model to work for two classes. In this case, have to train the model with a new output layer in place.

6. **Fine-tune your model:**

   One method of improving the performance is fine-tuning. Fine-tuning involves unfreezing some part of the base model and training the entire model again on the whole dataset at a very low learning rate. The low learning rate will increase the performance of the model on the new dataset while preventing overfitting [28].

Chapter 4

proposes Framework

4.1 Overview

The system aims to develop an automated system for agriculture that is useful for farmers, agricultural companies and agricultural engineers via performing the main agricultural tasks accurately to save time and effort by using deep learning to solve the problems faced by farmers such as:

1. Crop identification
2. Identification of diseases
3. Determining the stage of maturity to help determine the quantity of the crop for local use and the quantity of the crop for export.

Early detection of the disease helps farmers to make the right decision about treatment and preventive things such as:

- Fertilizers.
- Herbicides.
- Insecticides.

To increase the production of the crop. Figure (4.1) shows the proposed plant identificatoin and diseases detection system using Inception-v3.

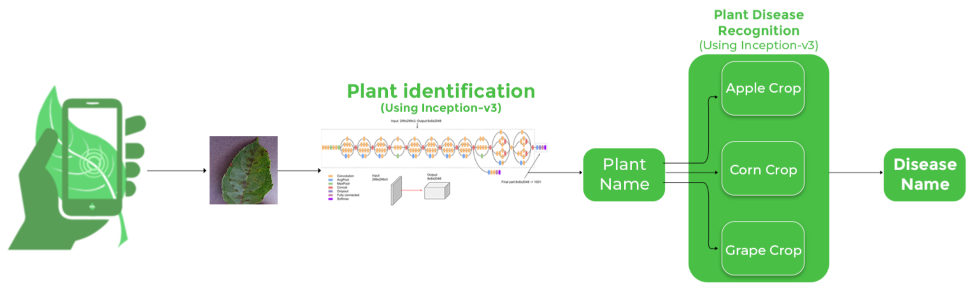

Figure 4.1: Plant Identificatoin and Diseases Detection Using Inception-v3.

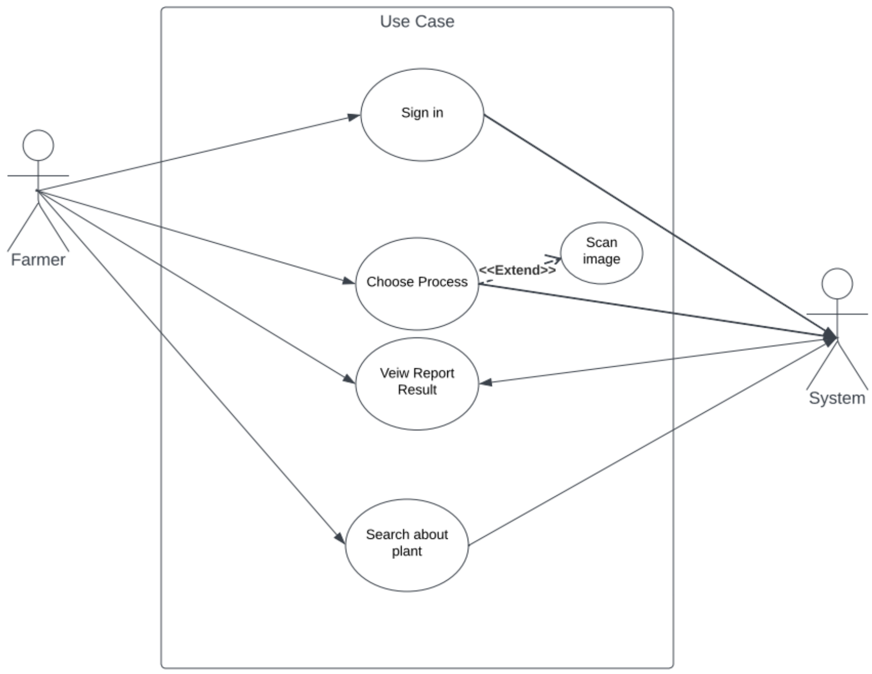

Figure 4.2: System Use Case Diagram.

4.2 Sign in Process

Logon is the procedure used to get access to application. Usually a logon requires that the user have (1) a user ID and (2) a password. Often, the user ID must conform to a limited length. The user ID can be freely known and is visible when entered at a keyboard or other input device. The password must be kept secret (and is not displayed as it is entered). Figure (4.3) shows the use case diagram of sign up and registration process.

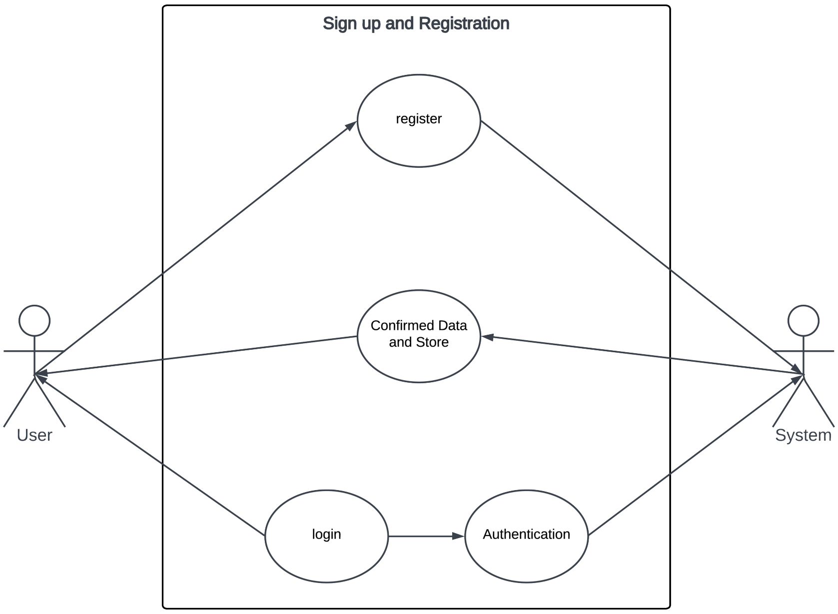

Figure 4.3: Sign Up and Registration Use Case Diagram.

Table (4.1) decribes the "Sign Up" Use Case Scenario.

Use case name:	Sign Up	
Actor(s):	Farmer	
Description:	This use case describes sign up steps to our Application.	
Typical of Events:	**Actor action**	**System Response**
	Step 1: fill registration Form without skip any field - Then Click ok.	Step 2: confirm with message in your e-mail.
Alternative:	Step 3: show massage error when massing data.	
Precondition:	No Precondition.	
Post condition:	Enter your profile or home page.	

Table 4.1: "Sign Up" Use Case Scenario.

Table (4.2) decribes the "Login" Use Case Scenario.

Use case name:	Log in	
Actor(s):	Farmer	
Description:	This use case describes Login steps Application.	
Typical of Events:	**Actor action**	**System Response**
	Step 1: logging to website enter username and password.	Step 2: check your user-name and password.
Alternative:	Step 3: check your username and password if it error.	
Precondition:	No Precondition.	
Post condition:	Enter your profile or home page.	

Table 4.2: "Login" Use Case Scenario.

Figure (4.4) shows Login Activity Diagram.

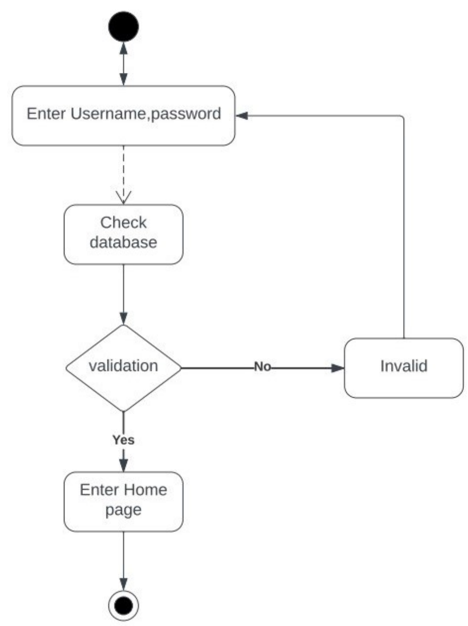

Figure 4.4: Login Activity Diagram.

4.3 Choose Process

Figure (4.5) shows Choose Process Use Case Diagram.

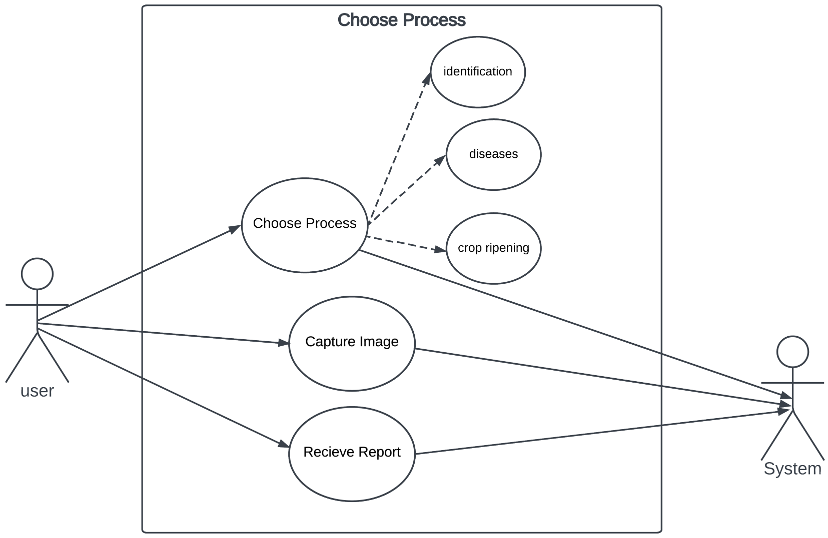

Figure 4.5: Choose Process Use Case Diagram.

Table (4.3) decribes the "Choose Process" Use Case Scenario.

<table>
<tr><td>Use case name:</td><td colspan="2">Choose Process</td></tr>
<tr><td>Actor(s):</td><td colspan="2">Farmer</td></tr>
<tr><td>Description:</td><td colspan="2">This use case describes the process that how to choose Process depend What does Farmer want to find out.</td></tr>
<tr><td rowspan="2">Typical of Events:</td><td>Actor action</td><td>System Response</td></tr>
<tr><td>Step 1: this use Application to Choose Process.</td><td>Step 2: Open Camera to scan image.</td></tr>
<tr><td>Alternative:</td><td colspan="2">Step 3: Choose Image from gallery</td></tr>
<tr><td>Precondition:</td><td colspan="2">No Precondition.</td></tr>
<tr><td>Post condition:</td><td colspan="2">Response report from disease Page.
Record operation in database.</td></tr>
</table>

Table 4.3: "Choose Process" Use Case Scenario.

4.3.1 Plants Recognition

The first process in the system is to determine the type of plant. Which is done by taking a picture of your plant, upload it, scan it and let the system identify it by performing a preprocessing process on the image and extracting its features and entering it into the deep learning model with Inception v3, and view the result in seconds. Figure (4.6) shows The proposed plant identification system.

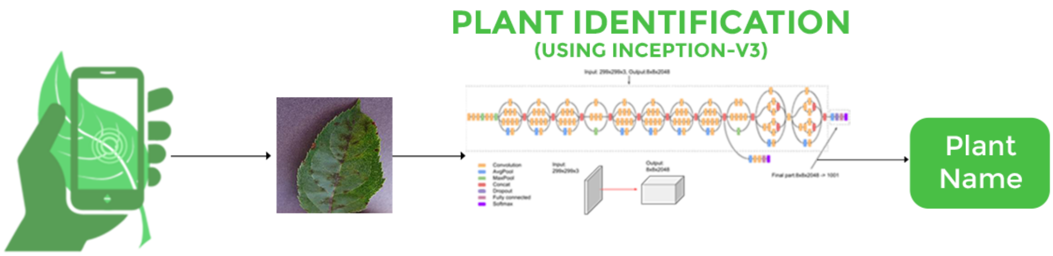

Figure 4.6: Plant Identification Model.

4.3.2 Plants Diseases Recognition

The second process in the system is plants diseases recognition because Plant disease recognition is very critical for agriculture due to its importance for increasing crop production. Which is done by taking a picture of your plant, upload it, scan it and let the system to find out whether this plant has a disease or not, and if it has a disease what kind of disease by performing a preprocessing process on the image and extracting its features and entering it into the deep learning model with Inception v3, and view the result in seconds. Figure (4.7) shows The proposed plant diseases recognition system.

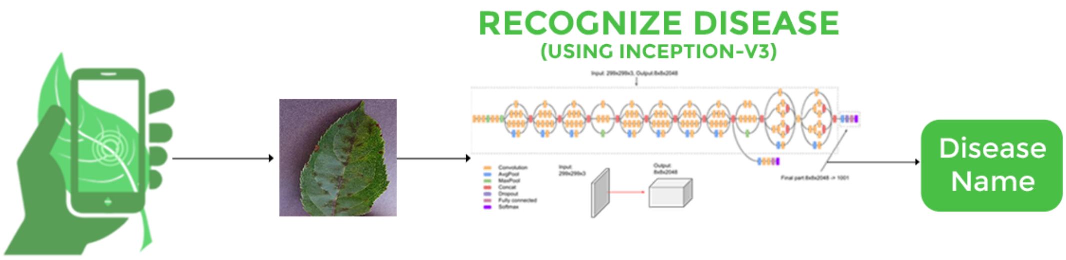

Figure 4.7: Plants Diseases Recognition.

4.3.3 Ripening Stage

The third process in the system is recognition ripening stage to know the time of harvesting the crop for local production and external export. This is done by taking a picture of your plant, upload it, scan it and allows the system to know the stage of ripening by performing a preprocessing process on the image and extracting its features and entering it into the deep learning model with Inception v3 and view the result in seconds. Figure (4.8) shows The proposed plant Ripeness rating system.

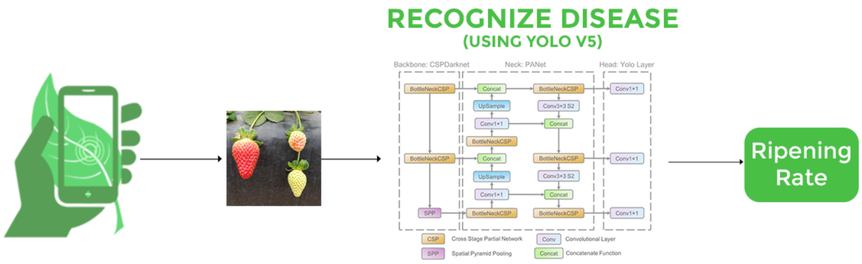

Figure 4.8: Plants Ripeness Rating System.

4.4 Search Process

Figure (4.9) shows search process sequence diagram.

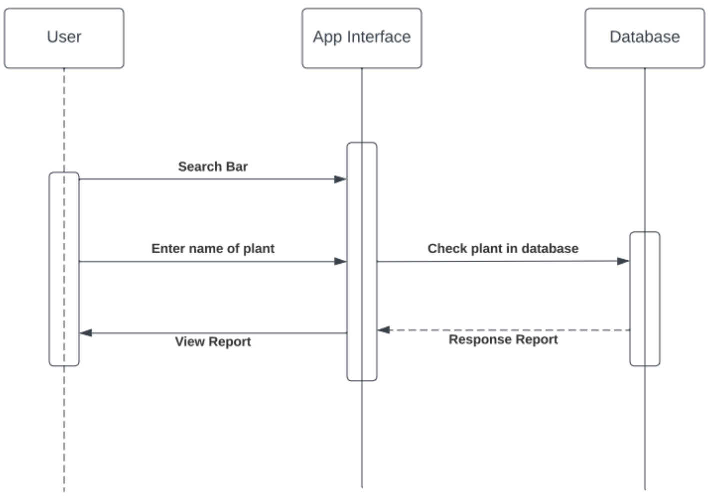

Figure 4.9: Search Process Sequence Diagram.

Figure (4.10) shows search process activity diagram.

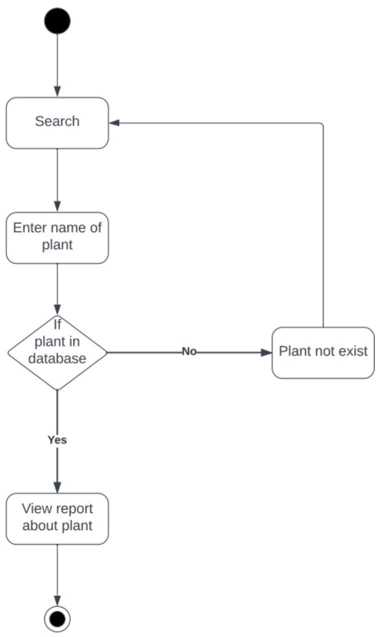

Figure 4.10: Search Process Activity Diagram.

Figure (4.11) shows search process use case diagram.

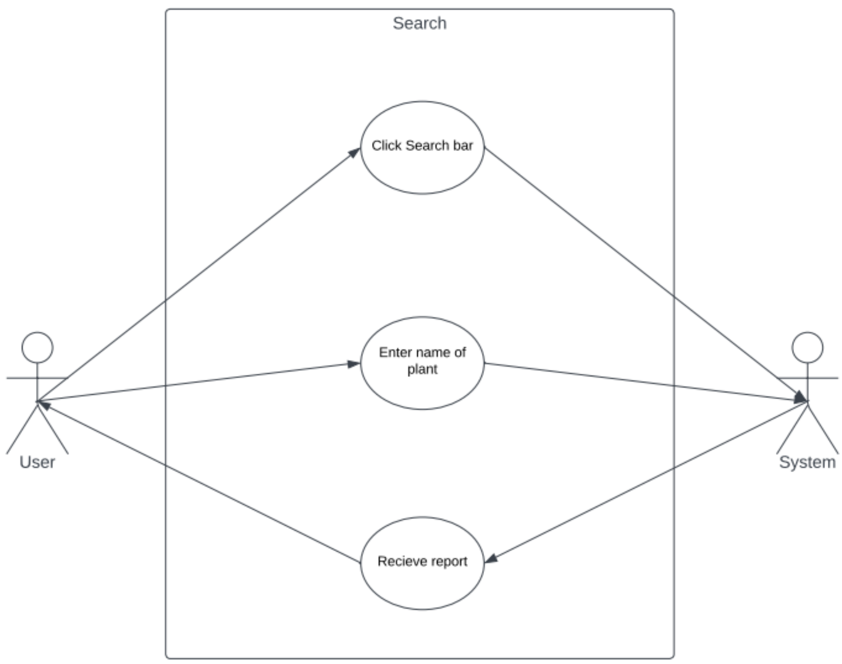

Figure 4.11: Search Process Use Case Diagram.

This system allows the user to search for a plant and know information about it by entering the name of the plant in the field search bar and the data for this plant will be displayed. Table (4.4) describes "Search" use case scenario.

Use case name:	Search	
Actor(s):	Farmer	
Description:	This use case describes the process that how search about specific plant.	
Typical of Events:	**Actor action**	**System Response**
	Step 1: this use Application to search about specific plant Enter name of plant.	Step 2: check in database if the plant is exist or not.
Alternative:	Step 3: Use identification plant and capture plant.	
Precondition:	No Precondition.	
Post condition:	Response report about plant.	

Table 4.4: "Search" Use Case Scenario.

Chapter 5

Experimental Results and Comparative Analysis

5.1 Experiment Specifications and Used Materials

This section presents and discusses all the details related to the experiments carried out to investigate and evaluate the performance of the proposed approaches. In this project, simulation experiments were performed on Google Colab with K80 GPU and 12 GB memory and a 16 GB RAM, Intel Core i7-4610M CPU (3.00 GHz, 1600 MHz, 4 MB L3 Cache, 2 cores, 37W). The proposed approach is designed with Tensorflow, Keras, Pytorch using Python.

5.2 Evaluation Metrics

To evaluate the performance of the proposed system, several performance metrics, namely, Accuracy, Recall (Sensitivity) and Precision, were calculated according to equations (5.1)-(5.2)-(5.3), respectively. where, TP is the true positive values, FP is the false positive values, TN is the true negative values, FN is the false negative values, and N is the total number of observations.

$$\text{Accuracy} = (TP + TN) / (TP + FN + TN + FP) \tag{5.1}$$

$$\text{Recall (Sensitivity)} = TP \,/\, (TP + FN) \tag{5.2}$$

$$\text{Precision} = TP \,/\, (TP + FP) \tag{5.3}$$

TP, **FP**, **TN**, and **FN** terms are defined as follows:

- **True Positive (TP):** the image is X and is classified as a X.
- **False Positive (FP):** the image is Y and is classified as a X.
- **True Negative (TN):** the image is Y and is classified as Y.
- **False Negative (FN):** the image is X and is classified as Y.

5.3 Results of The System

5.3.1 Dataset description

PlantVillage dataset is used to perform plant disease detection and Plants Identification. This dataset consists of 38 classes of different plants with total of 54303 healthy and unhealthy leaf images. It is also openly available on the internet. Each class contains approximately 2000 images. For every plant's health as well as diseased images of leaves are available. Figure (5.1) shows samples of healthy and unhealthy apple leaf. Most of the images belong to Tomato and Apple plants. The description of classes used in the system from this dataset is given in the table (5.1).

(a) Healthy

(b) Unhealthy

Figure 5.1: Samples of Healthy and Unhealthy Apple Leaf.

Table 5.1: Datasets (PlantVillage) description.

Dataset of plant disease detection and Plants Identification				
class	**Plant Name**	**Healthy or Diseased**	**Disease Name**	**Images (Number)**
C_0	Apple	Diseased	Apple_scab	1,890
C_1	Apple	Diseased	Black_rot	1,863
C_2	Apple	Diseased	Cedar_apple_rust	825
C_3	Apple	Healthy	-	4,935
C_4	Cherry_(including_sour)	Diseased	Powdery_mildew	3,156
C_5	Cherry_(including_sour)	Healthy	-	2,562
C_6	Corn_(maize)	Diseased	Cercospora_leaf_spotGray_leaf_spot	1,539
C_7	Corn_(maize)	Diseased	Common_rust	3,576
C_8	Corn_(maize)	Diseased	Northern_Leaf_Blight	2,955
C_9	Corn_(maize)	Healthy	-	3,486
C_10	Grape	Diseased	Black_rot	3,540
C_11	Grape	Diseased	Esca_(Black_Measles)	4,149
C_12	Grape	Diseased	Leaf_blight (Isariopsis_Leaf_Spot)	3,228
C_13	Grape	Healthy	-	1,269
C_14	Peach	Diseased	Bacterial_spot	6,891
C_15	Peach	Healthy	-	1,080
C_16	Pepper_bell	Diseased	Bacterial_spot	2,991
C_17	Pepper_bell	Healthy	-	4,434
C_18	Potato	Diseased	Early_blight	3,000
C_19	Potato	Diseased	Late_blight	3,000
C_20	Potato	Healthy	-	456
C_21	Strawberry	Diseased	Leaf_scorch	3,327
C_22	Strawberry	Healthy	-	1,368
C_22	Strawberry	Healthy	-	1,368
Dataset of ripeness assessment				
Real Dataset of Strawberry			Number of images = 310	

5.3.2 Results and Discussion

The proposed approaches were implemented considering three scenarios:

- Scenario I: Plants identification.
- Scenario II: Plant disease detection.
- Scenario III: Ripeness assessment.

Experimental results of the first two scenarios using both fine-tuned pre-trained Inception V3 and fine-tuned pre-trained VGG-16 models will be discussed in this section. The proposed Inception V3 model is trained using hyper-parameters (batch size = 64, epoch = 12). The VGG-16 model is trained using hyper-parameters (batch size = 32, epoch = 20). Experimental results of the third scenario using YOLOV5 model to recognize the ripeness stages in our dataset. The proposed YOLOV5 model is trained using hyper-parameters (batch size = 64, epoch = 150).

Scenario I: Plants Identification

This scenario presents plants identification model using raw data. The results of applying Inception V3 and VGG-16 fine-tuning procedures models on the PlantVillage dataset is discussed in this subsection. Table (5.2) shows the different performance metrics on dataset of the first scenario.

PlantVillage Dataset				
Model	**Accuracy**	**Recall**	**Precision**	**F - measure**
VGG-16 fine-tuning	98.2%	0.91	0.92	0.91
Inception-V3	100%	0.99	0.99	0.99

Table 5.2: Performance metrics for dataset of the Plants Identification model trained.

Table (5.2) shows the performance of the Inception V3 and VGG-16 fine-tuning procedures models for classification considering PlantVillage Dataset. For Inception V3 with PlantVillage Dataset, it is noticed that the obtained accuracy is 100%. However, the obtained recall percentage is 99.00%. On the other hand, for VGG-16 fine-tuning with PlantVillage Dataset, it is noticed that the obtained accuracy is 98.2%. However, the obtained recall percentage is 91.0%, where the percentage of correctly recognized plant images against the total number of actual plant images has been increased and the performance is improved accordingly. It also noticed that the use of Inception V3 using PlantVillage Dataset increases the precision to 99.00%. Accordingly, it is concluded that the Inception V3 model in general improves the accuracy by 1.8% and the F - measure by 8% compared to the proposed VGG16 fine-tuning model on the dataset. Figure (5.2) shows Scenario I: Plant Confusion Matrix.

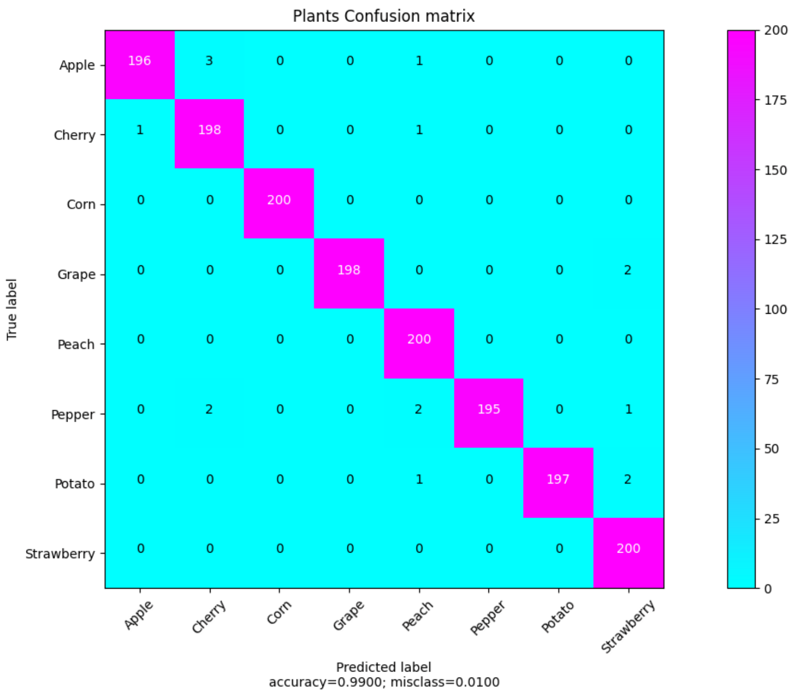

Figure 5.2: Scenario I: Plant Confusion Matrix.

Figure (5.3) shows Examples of Plants.

Figure 5.3: Examples of Plants.

Scenario II: Plant disease detection

This scenario presents Plant disease detection Model using raw data. The results of applying Inception V3 on the PlantVillage dataset is discussed in this subsection. Table (5.3) shows the performance metrics of the eight models on dataset of the second scenario.

PlantVillage Dataset					
Model	**Plant Name**	**Accuracy**	**Recall**	**Precision**	**F - measure**
Inception V3	Strawberry	100%	1	1	1
	Potato	100%	0.990	0.990	0.990
	Pepper	100%	0.995	0.995	0.995
	Peach	100%	0.995	0.995	0.995
	Grape	100%	0.993	0.993	0.993
	Cherry	100%	1	1	1
	Apple	100%	0.995	0.995	0.995
	Corn	100%	0.983	0.983	0.983

Table 5.3: Performance metrics for dataset of the Plant disease detection model trained.

Table (5.3) shows the performance of the Inception V3 model for Plant disease detection considering PlantVillage Dataset. For Inception V3 with PlantVillage Dataset after applying augmentation in it to have balanced data for increase the performance. Figures (5.4 to 5.11) show plant diseases confusion matrix.

1. In the first model "Strawberry disease classification" obtained in 4800 images (80% train and 20% test).

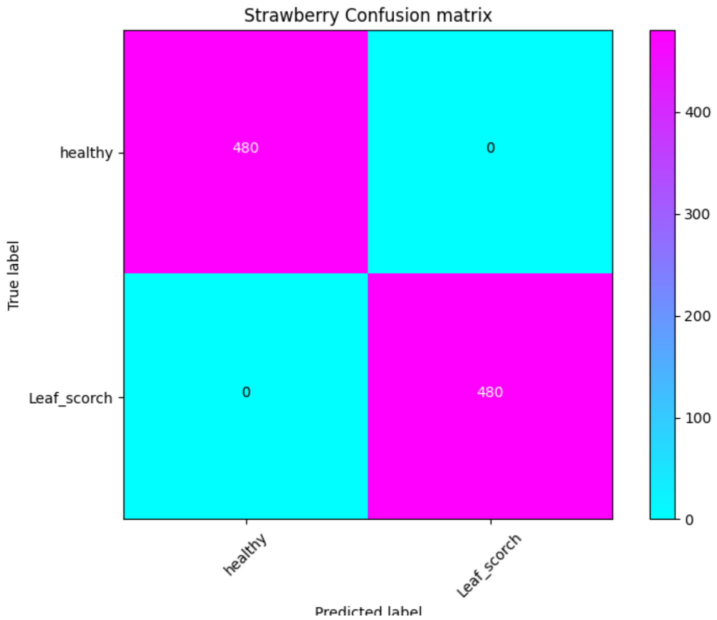

Figure 5.4: Strawberry Confusion Matrix.

2. In the second model "Potato disease classification" obtained in 7200 images (80% train and 20% test).

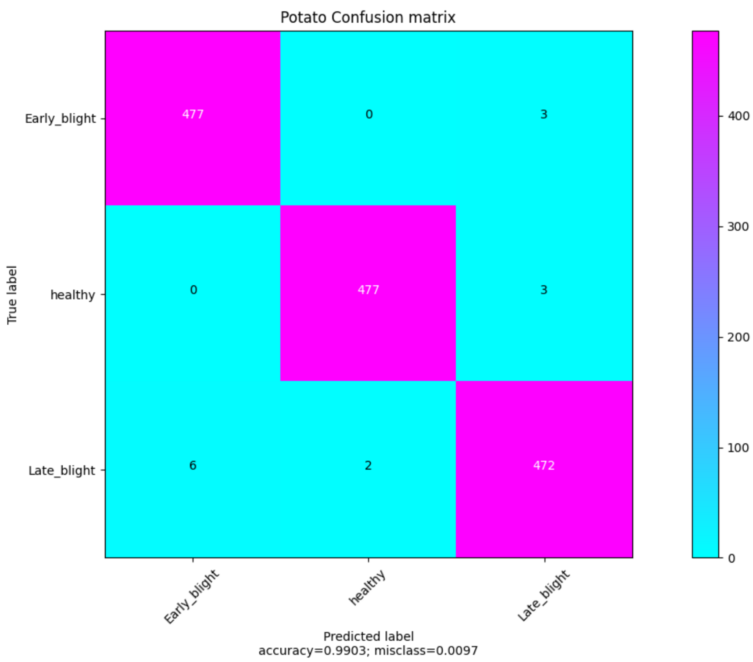

Figure 5.5: Potato Confusion Matrix.

3. In the third model "Pepper disease classification" obtained in 4800 images (80% train and 20% test).

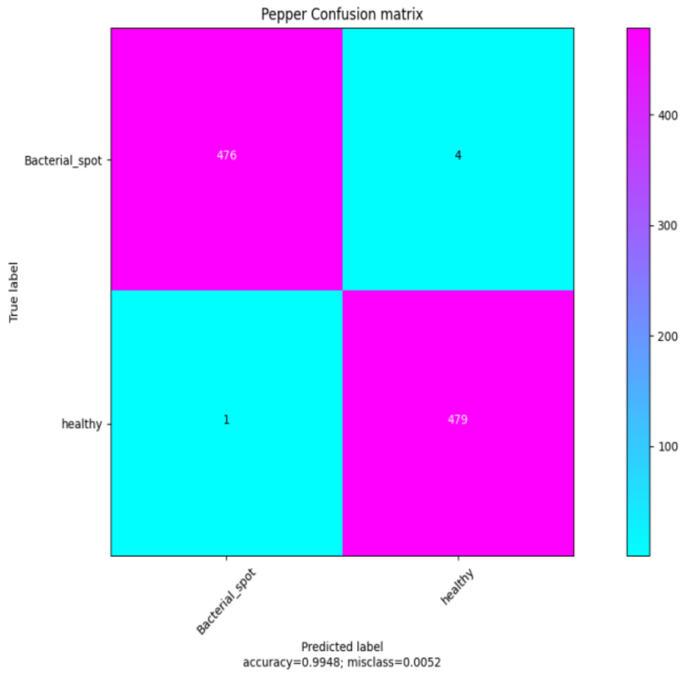

Figure 5.6: Pepper Confusion Matrix.

4. In the fourth model "Peach disease classification" obtained in 4800 images (80% train and 20% test).

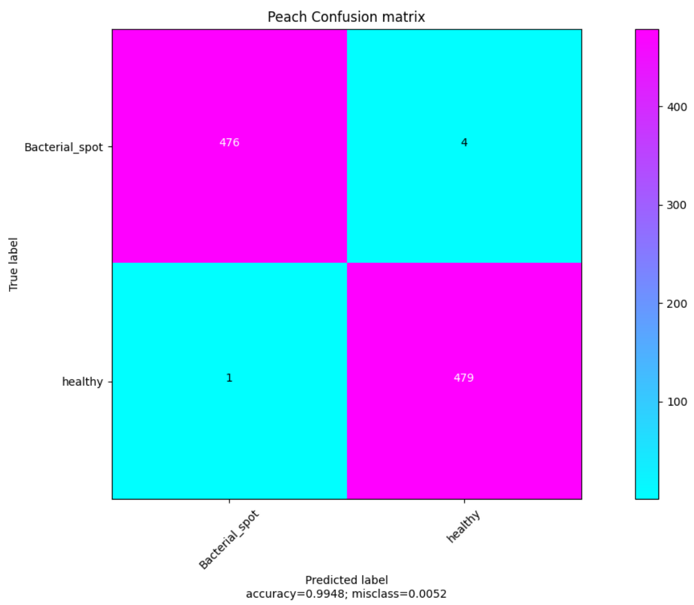

Figure 5.7: Peach Confusion Matrix.

5. In the fifth model "Grape disease classification" obtained in 9600 images (80% train and 20% test).

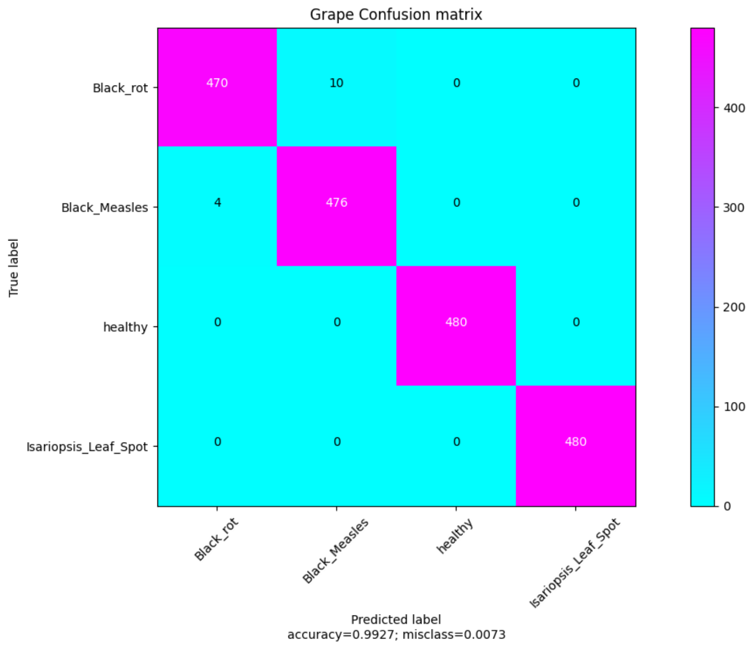

Figure 5.8: Grape Confusion Matrix.

6. In the sixth model "Cherry disease classification" obtained in 4800 images (80% train and 20% test).

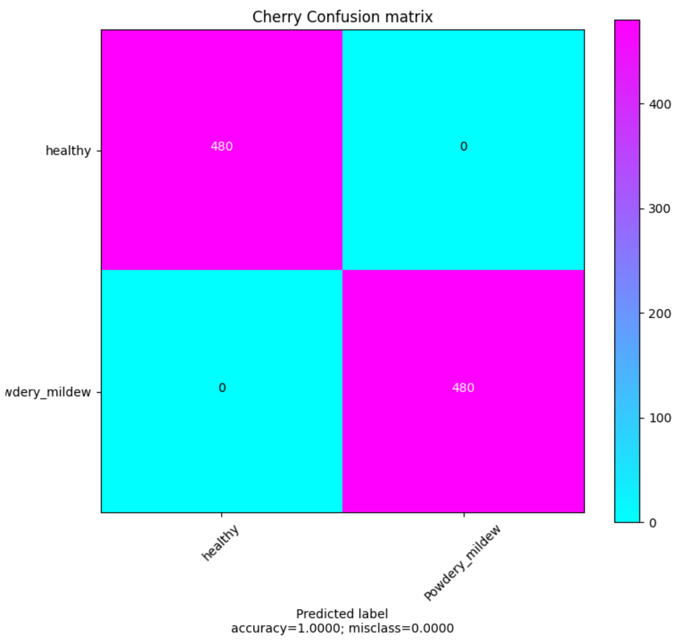

Figure 5.9: Charry Confusion Matrix.

7. In the seventh model "Apple disease classification" obtained in 7440 images (80% train and 20% test).

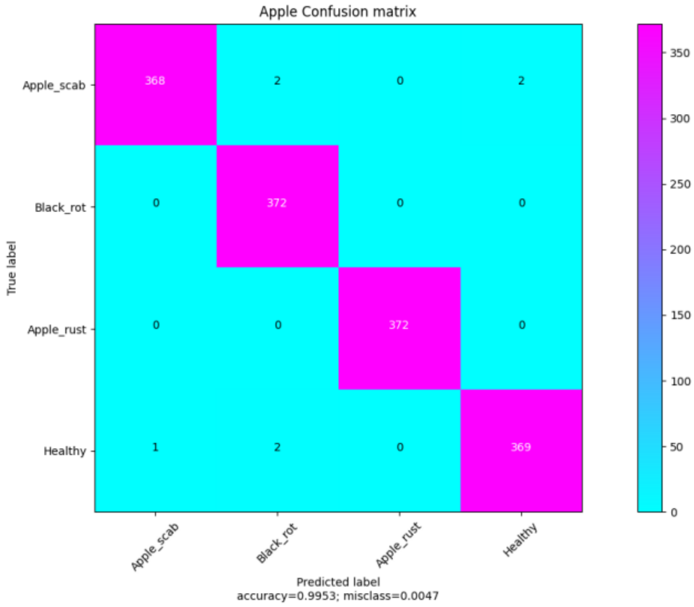

Figure 5.10: Apple Confusion Matrix.

8. In the eight model "Corn disease classification" obtained in 9600 images (80% train and 20% test).

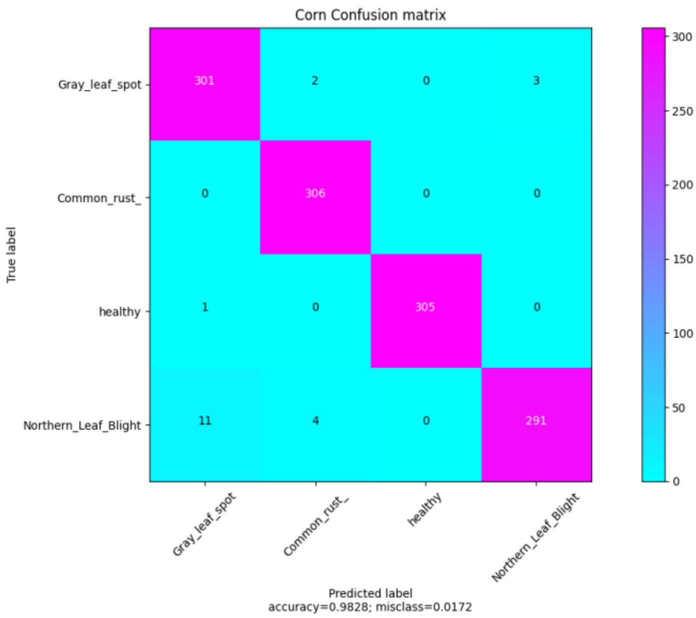

Figure 5.11: Corn Confusion Matrix.

Figure (5.12) shows Examples of Plant Diseases.

Figure 5.12: Examples of Plant Diseases.

Scenario III: Ripeness assessment

This scenario presents ripeness assessment model using raw data. The results of object detection training using YOLOv5 on the real dataset that consist of 310 images of Strawberry is discussed in this subsection. Table (5.4) shows the performance metrics Ripeness Assessment Model on this dataset of the third scenario. Figure (5.13) shows Results of 'feature extraction' Training.

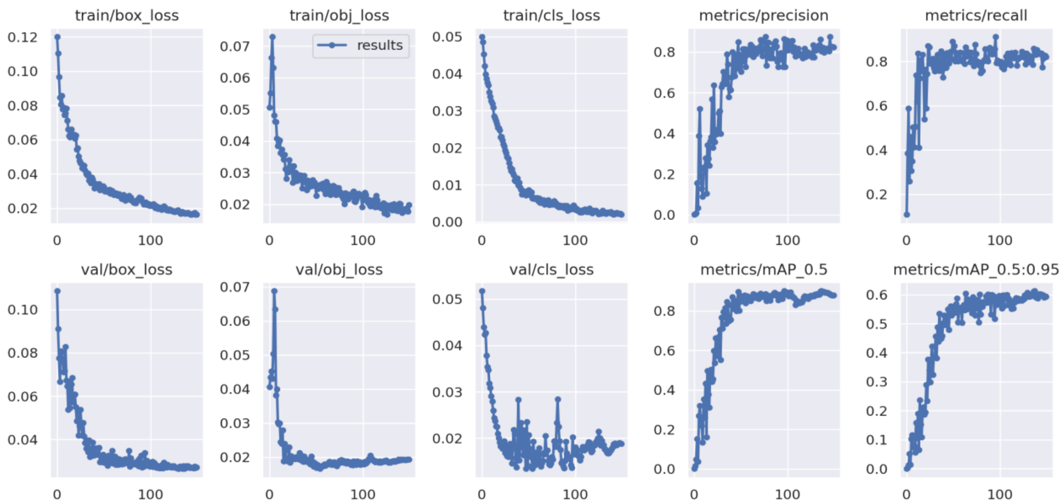

Figure 5.13: Results of 'feature extraction' Training.

Real Dataset of Strawberry				
Model	**Class**	**Recall**	**Precision**	**mAP.5**
YOLO-V5	all	0.902	0.855	0.912
	green	0.915	0.841	0.905
	pink	0.875	0.951	0.955
	red	0.933	0.767	0.876
	white	0.886	0.861	0.914

Table 5.4: Performance metrics for dataset of Ripeness assessment model.

Table (5.4) shows the performance of YOLOv5 of Object Detection model for classification considering Real dataset of Strawberry . For YOLOv5 with real dataset of Strawberry, it is noticed that the obtained accuracy is 87.8%, the obtained recall percentage is 90.2%, where the percentage of correctly recognized plant images against the total number of actual plant images and the precision 85.5%. Figure (5.14) shows Ripeness Confusion Matrix.

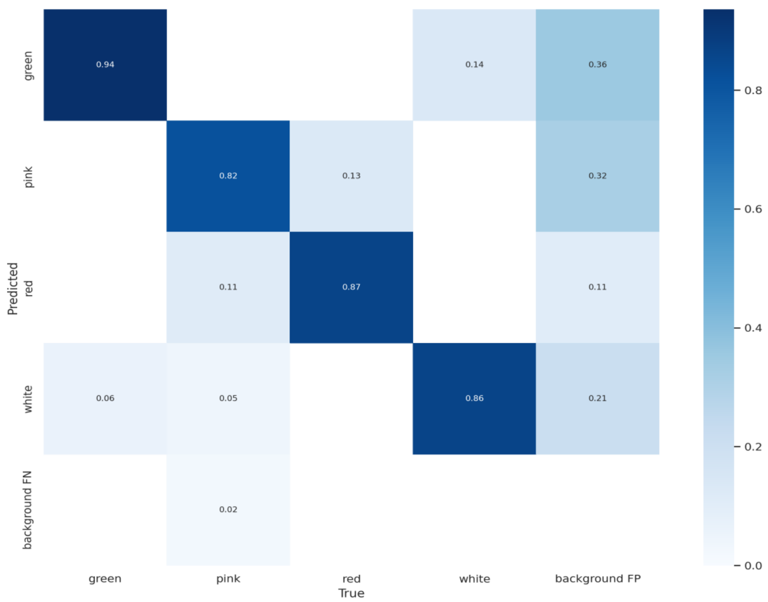

Figure 5.14: Ripeness Confusion Matrix.

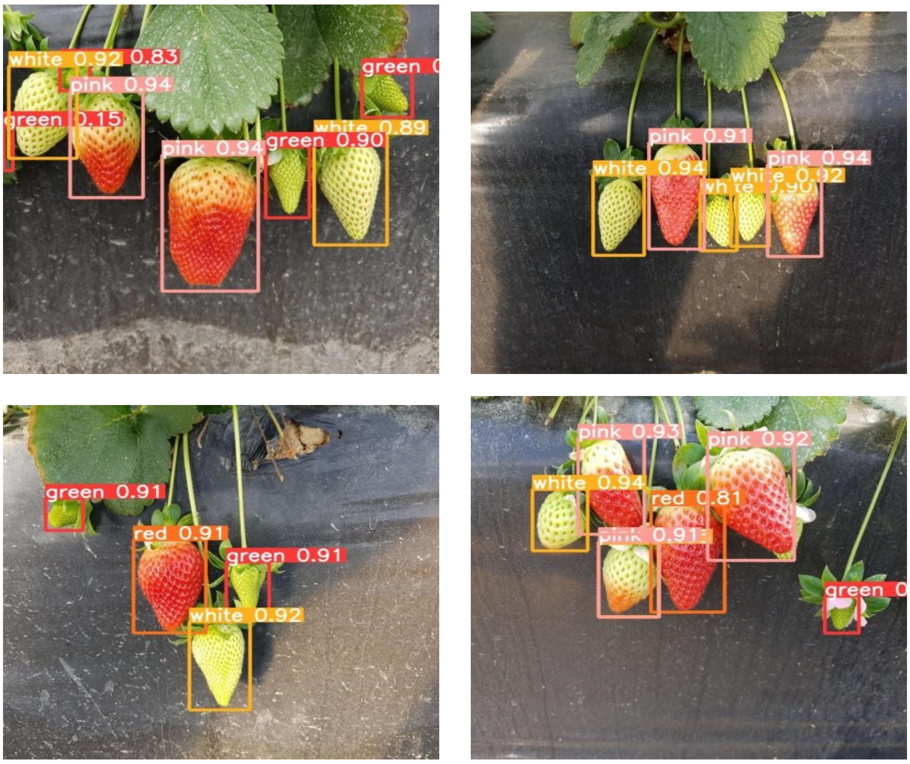

Figure 5.15: Examples of Object Detection.

Chapter 6

Conclusions & Future Work

6.1 Conclusions

There are many developed methods for the detection and classification of plant diseases using the leaves of plants. However, there is still no efficient and effective commercial solution that can be used to identify diseases and plants identification. NABTA used (InceptionV3) for the detection of plant identification and plant diseases using healthy- and diseased-leaf images of plants.

To train and test the model, we used the standard PlantVillage dataset with 54,303 images, which were all captured in laboratory conditions. This dataset consists of 38 classes of different plants, we used a dataset with 4800 images for cherry, 4800 for peach, 4800 for Pepper, 7200 for Potato, 4800 for Strawberry, 7440 for Apple, 9600 for Grape, 9600 for Corn, after splitting the dataset into 80 – 20 (80% for training, 20% for testing), NABTA achieved the best accuracy rate of

99.52% in the InceptionV3 model for apple,

99.9% in the InceptionV3 model for cherry,

99.27% in the InceptionV3 model for grape,

99.4% in the InceptionV3 model for peach,

98.28% in the InceptionV3 model for corn,

99% in InceptionV3 model for potato,

99.4% in the InceptionV3 model for Pepper,

and **99.9%** in the InceptionV3 model for strawberries.

Moreover, NABTA allows the user to search for a plant and know information about it by entering the name of the plant in the field search bar and the data for this plant will be displayed.

Finally, NABTA can classify the ripeness stages of strawberry. The proposed system has three main stages; pre-processing, feature extraction, and ripeness classification. which uses Object detection to allow us to identify and locate objects in an image and determine and track their precise locations, all while accurately labeling them, The proposed classification approach was implemented by applying to resize and extracting color components for each image. Then, feature extraction was applied to each pre-processed image, this dataset consists of 4 classes of different stages of ripeness (Green, white, pink, and green) Yolov5 model is developed for ripeness stage classification. For Strawberry ripeness assessment, based on the obtained experimental results, the highest ripeness mAP of 91.2%.

6.2 For Future work

We aim to:

1. Increase the number of plants in our application.
2. Inserting the robotics part for automatic plant care.
3. Detection and estimation of weeds.
4. Suggesting Fertilizers, Herbicides, and Insecticides for early detection of the disease for speedy treatment.
5. Follow the stages of plant growth.
6. Add an export user to manage data and model.
7. Add System to care with plants.

Bibliography

[1] Dries, L., Heijman, W., Jongeneel, R., Purnhagen, K., & Wesseler, J. (Eds.). (2019). *EU Bioeconomy Economics and Policies: Volume II. Cham*, UK: Springer International Publishing.

[2] J. Brezmes, E. Llobet, X. Vilanova, G. Saiz, and X. Correig, Fruit ripeness monitoring using an electronic nose, *Sensors and Actuators B-Chem Journal*, vol.69, no.3, pp.223–229, Elsevier, 2000.

[3] Z. May and M. H. Amaran, Automated ripeness assessment of oil palm fruit using RGB and fuzzy logic technique", *InProc. of the 13th WSEAS international conference on Mathematical and computational methods in science and engineering (MACMESE'11),* Stevens Point, Wisconsin, USA, pp.52–59, World Scientific and Engineering Academy and Society (WSEAS), 2011.

[4] Mahmudul Hassan, S. K., & Kumar Maji, A. (2021). Identification of plant species using deep learning. *In Proceedings of International Conference on Frontiers in Computing and Systems* (pp. 115-125). Springer, Singapore

[5] B. K. Varghese, A. Augustine, J. M. Babu, D. Sunny, and E. S. Cherian. *Infoplant: Plant recognition using convolutional neural networks.* In 2020 Fourth International Conference on Computing Methodologies and Communication (ICCMC), pages 800807. IEEE, 2020.

[6] S. A. Riaz, S. Naz, and I. Razzak. *Multipath deep shallow convolutional networks for large scale plant species identification in wild image.* In 2020 International Joint Conference on Neural Networks (IJCNN), pages 17. IEEE, 2020.

[7] Kaya, A., Keceli, A. S., Catal, C., Yalic, H. Y., Temucin, H., & Tekinerdogan, B. (2019). *Analysis of transfer learning for deep neural network based plant*

classification models. Computers and electronics in agriculture, 158, pages 20–29.

[8] Manasa, N., Herur, P., Shetty, P., Prarthana, N., & Venkatrao, P. (2019, June). *Plant recognition using watershed and convolutional neural network.* In 2019 3rd International conference on Electronics, Communication and Aerospace Technology (ICECA) (pp. 969–973). IEEE.

[9] Beikmohammadi, A., & Faez, K. (2018, December). *Leaf classification for plant recognition with deep transfer learning.* In 2018 4th Iranian Conference on Signal Processing and Intelligent Systems (ICSPIS) (pp. 21–26). IEEE.

[10] Wang, P., Niu, T., Mao, Y., Zhang, Z., Liu, B., & He, D. (2021). *Identification of Apple Leaf Diseases by Improved Deep Convolutional Neural Networks With an Attention Mechanism.* Frontiers in Plant Science, 1997.

[11] Yu, H., Liu, J., Chen, C., Heidari, A. A., Zhang, Q., Chen, H., ... & Turabieh, H. (2021). *Corn leaf diseases diagnosis based on K-means clustering and deep learning.* IEEE Access, 9, 143824-143835.

[12] Wang, G., Yu, H., & Sui, Y. (2021). *Research on Maize Disease Recognition Method Based on Improved ResNet50.* Mobile Information Systems, 2021.

[13] Ji, M., Zhang, L., & Wu, Q. (2020). *Automatic grape leaf diseases identification via UnitedModel based on multiple convolutional neural networks.* Information Processing in Agriculture, 7(3), 418–426.

[14] Thakur, R., Suryawanshi, G., Patel, H., & Sangoi, J. (2020, May). *An innovative approach for fruit ripeness classification.* In 2020 4th International Conference on Intelligent Computing and Control Systems (ICICCS) (pp. 550-554). IEEE.

[15] Shao, Y., Wang, Y., Xuan, G., Gao, Z., Hu, Z., Gao, C., & Wang, K. (2020). *Assessment of strawberry ripeness using hyperspectral imaging.* Analytical Letters, 54(10), 1547-1560.

[16] Ünal, Z. (2020). *Smart farming becomes even smarter with deep learning—a bibliographical analysis.* IEEE Access, 8, 105587-105609.

[17] Hassoon, I. M., Kassir, S. A., & Altaie, S. M. (2018). *A review of plant species identification techniques.* Int Jour Sci Res, 7(8), 2319-7064.

[18] Mohanty, S. P., Hughes, D. P., & Salathé, M. (2016). *Using deep learning for image-based plant disease detection.* Frontiers in plant science, 7, 1419.

[19] Masmoudi, I., & Lghoul, R. (2021). *A Deep Convolutional Neural Network Approach for Plant Leaf Segmentation and Disease Classification in Smart Agriculture.* In Intelligent Computing (pp.1044–1055). Springer, Cham.

[20] Esraa El-hariri. PhD Fayoum Universty (2020). *Prediction_Technique_for_Historical_Heritage_Preservation_Based_on_IoT-8-3-166-1.*

[21] Alzubaidi, L., Zhang, J., Humaidi, A. J., Al-Dujaili, A., Duan, Y., Al-Shamma, O., ... & Farhan, L. (2021). *Review of deep learning: Concepts, CNN architectures, challenges, applications, future directions.* Journal of big Data, 8(1), 1-74.

[22] Rathi, D. (2018). *Optimization of transfer learning for sign language recognition targeting mobile platform.* arXiv preprint arXiv:1805.06618.

[23] Sam, S. M., Kamardin, K., Sjarif, N. N. A., & Mohamed, N. (2019). *Offline signature verification using deep learning convolutional neural network (CNN) architectures GoogLeNet inception-v1 and inception-v3.* Procedia Computer Science, 161, 475-483.

[24] Barker, M. (2019). *Inception V3 Deep Convolutional Architecture For Classifying Acute Myeloid/Lymphoblastic Leukemia.* intel. com.

[25] Demir, A., Yilmaz, F., & Kose, O. (2019, October). *Early detection of skin cancer using deep learning architectures: resnet-101 and inception-v3.* In 2019 medical technologies congress (TIPTEKNO) (pp. 1–4). IEEE.

[26] Garg, A., & Tyagi, S. (2021). *IoT-a supportive system for solving issues related to medical monitoring.* Computational Intelligence and Predictive Analysis for Medical Science: A Pragmatic Approach, 6, 53.

[27] Ribani, R., & Marengoni, M. (2019, October). *A survey of transfer learning for convolutional neural networks.* In 2019 32nd SIBGRAPI conference on graphics, patterns and images tutorials (SIBGRAPI-T) (pp. 47–57). IEEE.

[28] `https://www.v7labs.com/blog/transfer-learning-guide`.