VISVESVARAYA TECHNOLOGICAL UNIVERSITY

JNANA SANGAMA CAMPUS, BELGAVI-590018

PROJECT REPORT

On

"DETECTION LACK OF NUTRIENTS IN COFFEE AND BANANA UING IMAGE PROCESSING"

Submitted by

AHIMSA JAIN	**4AL17IS001**
THANGSABAM BIKUMAR SINGH	**4AL17IS051**
ZEENAL MANOLA LOBO	**4AL17IS053**
VARADA	**4AL16IS058**

In partial fulfillment of the requirements for the degree of

BACHELOR OF ENGINEERING

In

INFORMATION SCIENCE AND ENGINEERING

Under the Guidance of

Mr. PRADEEP NAYAK

Assistant Professor

DEPARTMENT OF INFORMATION SCIENCE AND ENGINEERING

ALVAS INSTITUTE OF ENGINEERING AND TECHNOLOGY

Moodbidri-574225, Karnataka

2020– 2021

ALVA'S INSTITUTE OF ENGINEERING AND TECHNOLOGY

MIJAR, MOODBIDRI D.K. -574225

KARNATAKA

DEPARTMENT OF INFORMATION SCIENCE AND ENGINEERING

CERTIFICATE

*Certified that the project work entitled "**DETECTION LACK OF NUTRIENTS IN COFFEE AND BANANA UING IMAGE PROCESSING**" is a bonafide work carried out by*

AHIMSA JAIN	**4AL17IS001**
THANGSABAM BIKUMAR SINGH	**4AL17IS051**
ZEENAL MANOLA LOBO	**4AL17IS053**
VARADA	**4AL16IS058**

in partial fulfilment for the award of BACHELOR OF ENGINEERING in **INFORMATION SCIENCE AND ENGINEERING** of the **VISVESVARAYA TECHNOLOGICAL UNIVERSITY, BELGAUM** during the year 2020–2021. It is certified that all corrections/suggestions indicated for Internal Assessment have been incorporated in the report deposited in the departmental library. The project report has been approved as it satisfies the academic requirements in respect of project work prescribed for the Bachelor of Engineering Degree.

Mr. PRADEEP NAYAK	**Mr. SUDHEER SHETTY**	**Dr. PETER FERNANDES**
Project Guide	**Head of Department**	**Principal**

PRINCIPAL
Alva's Institute of Engg. & Technology
Mijar, MOODBIDRI - 574 225, D.K.

Name of the Examiners **Signature with Date**

1.

2.

ALVA'S INSTITUTE OF ENGINEERING AND TECHNOLOGY

MIJAR, MOODBIDRI D.K. -574225

KARNATAKA

DEPARTMENT OF INFORMATION SCIENCE & ENGINEERING

DECLARATION

We,

AHIMSA JAIN

THANGSABM BIKUMAR SINGH

ZEENAL MANOLA LOBO

VARADA

hereby declare that the dissertation entitled, "*DETECTION LACK OF NUTRIENTS IN COFFEE AND BANANA UING IMAGE PROCESSING*" is completed and written by us under the supervision of our guide **Mr. PRADEEP NAYAK**, Assistant Professor, Department of Information Science and Engineering, Alva's Institute of Engineering And Technology, Moodbidri, in partial fulfilment of the requirements for the award of the degree BACHELOR OF ENGINEERING in DEPARTMENT OF INFORMATION SCIENCE & ENGINEERING of the VISVESVARAYA TECHNOLOGICAL UNIVERSITY, BELAGAVI during the academic year 2020-2021. The dissertation report is original and it has not been submitted for any other degree in any university.

AHIMSA JAIN	4AL17IS001
THANGSABAM BEEKUMAR SINGH	4AL17IS051
ZEENAL MANOLA LOBO	4AL17IS053
VARADA	4AL16IS058

ACKNOWLEDGEMENT

The satisfaction and euphoria that accompany a successful completion of any task would be incomplete without the mention of people who made it possible, success is the epitome of hard work and perseverance, but steadfast of all is encouraging guidance.

So, with gratitude we acknowledge all those whose guidance and encouragement served as beacon of light and crowned the effort with success.

We thank our project guide **Mr. PRADEEP NAYAK,** Assistant Professor, in Department of Information Science & Engineering, who has been our source of inspiration. She has been especially enthusiastic in giving her valuable guidance and critical reviews.

The selection of this project work as well as the timely completion is mainly due to the interest and persuasion of our project coordinator **Dr. KIRAN B. MALAGI,** Professor, Department of Information Science & Engineering. We will remember their contribution for ever.

We sincerely thank, **Mr. SUDHEER SHETTY**, Associate Professor and Head, Department of Information Science & Engineering who has been the constant driving force behind the completion of the project.

We thank our beloved Principal **Dr. PETER FERNDES,** for his constant help andsupport throughout.

We are indebted to **Management of Alva's Institute of Engineering and Technology, Mijar, Moodbidri** for providing an environment which helped us in completing our project.

Also, we thank all the teaching and non-teaching staff of Department of Information Science & Engineering for the help rendered.

Ms. AHIMSA JAIN **USN: 4AL17IS001**

Mr. THANGSABAM BEEKUMAR SINGH **USN: 4AL17IS051**

Ms. ZEENAL MANOLA LOBO **USN: 4AL17IS053**

Ms. VARADA **USN: 4AL16IS058**

ABSTRACT

The main Aim of this project is to help the coffee and banana growing farmers to classify and recognize the nutrient deficiency in coffee and banana. Presently farmers taking help of the experts to take decision but it is again problem for the farmers because of consultancy fees and unavailability of experts. The project proposed farmers to take time to time decision. There are different nutrients, lack nutrient like Boron & Magnesium in Banana and Calcium and Phosphorous in Coffee can be detected and providing required nutrients to plants which will help in improving quality of the products.

The project proposed website, which the farmer can directly access. Front End consists of upload photo (testing data) & predicted result and Back End consists trained model, preprocessed testing data and compared both trained model and testing data.

To trained the model, Mobile camera is used to collect the images which is training datasets. The datasets were collected from Coffee and Banana farms. For Coffee Calcium & Phosphorous nutrient deficiency images were taken and for Banana Boron & Magnesium deficiency were taken. The system uses the image processing technique to trained model, Image processing is done by several process such as Collecting the training datasets, Pre-processing data & converting the image data to array by OpenCV, Stored the array to Pickle module and Trained the model by CNN algorithm. Then the model will stored in tensorflow.keras.model.

The farmers need to captured the photo and upload to the website. Then the upload image will be converted into array & preprocess. After preprocess the image will compare with trained model by tensorflow.keras(predict). And the result (about lack nutrient and correct measures) will be display in same page.

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

OVERVIEW

Coffee is known as the most important crop commodity in the world. Banana is grown in almost all regions and plays a key role in economies of many developing countries including India. Coffee is known as the most important crop commodity in the world. People around the world drink two billion cups of coffee every day. A total of 25 million of families depend for coffee for living specifically for business or as growers. For the last 15 years, 43 percent of coffee consumption has arisen worldwide. Bananas are grown in almost all regions and play a key role in the economies of many developing countries including India. In terms of gross value of production, bananas are the world's fourth most important food crop after rice, wheat and maize. They are a staple food and an export commodity.

In the developing world, technologies play vital role in the agricultural field. Human survival mainly depends on Agriculture. We are still using traditional methods in agricultural practices. Identifying nutrient deficiency in crops is still difficult for farmers. If we manually identify nutrient deficiency in crops, it will take more time, more work is done and cost is also needed. The coffee and banana growing farmers face different problems due to the Pests, Diseases, Fungi, Climate Change, Unpredictable Rains and lack of nutrient deficiency is also one of them identification of nutritional deficiencies is done manually by the farmers or experts. Again this is problem for farmers because farmers need to wait for long time because of unavailability of experts and high consultation fees. So we proposed solution which is a model for detect and classify the banana and coffee which are having lack of nutrients.

The proposed system helps the farmers to do smart farming and allows the farmers to take time to time decision and also which improves the accuracy in classification. The process of detecting lack nutrient is done by image processing. Convolutional Neutral Networks (CNN) is used to extract the feature of lack nutrients. After features extraction, the proposed system uses Camera to collect the input image Data and upload to the project website, it detects the lack nutrient then as a result farmer will identify lack nutrient through website.

LITERATURE SURVEY

A literature review is a written document that presents a logically argued case founded on a comprehensive understanding of the current state of knowledge about a topic of study. This case establishes a convincing thesis to answer a study's question. It will also give readers the necessary background to understand the research work. User authentication is most fundamental component in computer security context. There are various techniques used for authenticating the user, such as recognition base authentication, recall based authentication, pure recalled based authentication. To overcome these drawbacks Hybrid images password were introduced. It is resistant to dictionary attack, bruite force attack etc

Classification and Detection of Nutritional Deficiencies in Coffee Plants Using Image Processing and CNN:

Khenilyn P. Lewis, Juancho D. Espineli [1] addresses the problem of "Classification and Detection of Nutritional Deficiencies In Coffee Plants Using Image Processing and CNN", The study covered the four varieties of coffee. Image processing technique was utilized in converting the images into grayscale to binary. And CNN is used to predict the nutritional deficiency in coffee plans through classification and detection. In the proposed system once the nutritional deficiency is found or identified, proto type will display recommended fertilizer for the plants. 1000 images with the 8 nutritional deficiencies is used (B, Ca, Fe, N, P, K, Mg, Z). The proposed system gave high accuracy for classification of coffee plants and the level of detection and classification depends upon the size of the leaves.

Identification of Plant Nutrient Deficiencies Using Convolutional Neural Networks:

Ukrit Watchareeruetai, Pavit Noinongyao, Chaiwat Wattanapaiboonsuk, Puriwat Khantiviriya [2] A novel image analysis method for identifying nutrient deficiencies in plant based on its leaf is proposed. First, the proposed method divides an input leaf image into small blocks. Second, each block of leaf pixels is fed to a set of convolutional neural networks (CNNs). Each CNN is specifically trained for a nutrient deficiency and is utilized to decide if a block is presenting any symptom of the corresponding nutrient deficiency. Next, the responses from all CNNs are integrated to produce a single response or the block using a winner-take-all strategy.

Finally, the responses from all blocks are integrated into one using a multi-layer perceptron to produce a final response for thewhole leaf. Validation of the proposed method was performed on a set of black gram(Vigna mungo) plants grown under nutrient controlled environments. Five types ofdeficiencies, i.e., Ca, Fe, K, Mg, and N deficiencies, and a group of plants with completenutrients were studied. A dataset consisting of 3,000 leaf images was collected and usedfor experimentation. Experimental results indicate the superiority of the proposed methodover trained humans in nutrient deficiency identification. Experimental results present thefeasibility of using CNNs to identify nutrient deficiencies in black gram plants based ontheir leaves, and indicate its superiority over humans in performance comparison.

Automatic Classification of Nutritional Deficiencies in Coffee Plants:

Diego Monsalve, Maria Trujillo, Deisy Chaves [3]. Classification of nutritional deficiencies in coffee plants". Classification of Nutritional deficiency in coffee plants is a problem for coffee farmers because farmers cannot identify the plants through their naked eyes so to classify we need to use automatic classification. Nutritional deficiency is caused due to the lack of macro-nutrient and micro-Nutrients. Such as B, Ca, Mg, Mn, K, P, N. The proposed approach for automatically classifying nutritional deficiency has the first step is image processing secondly visual feature are extracted by local and global descriptors. Thirdly the extracted features are used to build a random forest model to classify the nutritional deficiencies of the analyzed coffee leaf. The proposed model correctly classifies boron, nitrogen, calcium and phosphorous.

An Algorithm for Detection of Nutritional Deficiencies from Digital Images of Coffee Leaves Based on Descriptors and Neural Networks:

Jose Sosa, Janidet Ramirez, Luis Vives and Guillermo Kemper [4]. Classification of coffee plants which are having nutritional deficiencies is very difficult for the farmer through naked eye. So in this paper it uses a Random forest a Machine Learning technique based on decision tree it is used to classify automatically nutritional deficiencies in coffee plants. Then it uses scale – invariant feature extraction (SIFT) algorithm to extract local features .Based on the color and shape global features are defined.

PROBLEM STATEMENT

Nutrients play a major role in Agricultural field and crop production. Here we considered Coffee and Banana, because it is grown in almost all the states of the country. There are number of reasons for decreasing of coffee and banana production, one such problem faced by farmers is nutrient deficiency. It is difficult for farmers to identify which crops has lack of nutrients in their crop, so they consult with professional expert .so it is again a problem for the farmers because the consultation charges are high and it is also not possible to get a result on time. The proposed system gives the efficient result with high accuracy and efficiency. Detection of nutrient deficiency will improve the quality of the product.

EXISTING SYSTEM

Presently, there is no smart system to detect lack nutrients by farmer. Farmers need to meet the experts as per expert advices the farmer take the correction measure.

PROPOSED SYSTEM

Coffee and Banana growing Farmers has faced issue to search for expertise and also to wait time to meet expertise. The basis for our proposed system helps the farmers to do smart farming and allows the farmers to take time to time decision and also which improves the accuracy in classification. We contributed towards this claim in the following ways, we designed website, through which farmers can identify lack of nutrient in coffee and Banana just by uploading the pictures. The proposed system uses Camera to collect the input image Data and upload to the project website, it detects the lack nutrient then as a result farmer will identify lack nutrient through website.

The process of detecting lack nutrient is done by image processing. The shape and color feature are used to recognize and classify the lack of nutrients by using CNN algorithm.

SCOPE

The proposed framework helps to overcome the problems faced by the farmers. And it improves the growth of the plant and yield. And also it identifies the type of nutrient deficiency it is lack of so that farmers can contact the experts to know what the fertilizer to be used.

OBJECTIVE

- The proposed method is based on image processing which is used to identify the lack of nutrient like phosphorous & Calcium in coffee and Magnesium & Boron in banana.
- This project helps the Coffee and Banana growing farmers to efficiently classify and recognize lack of nutrient deficiencies.
- This project provides the efficient result with high accuracy and efficiency.
- This project proposes a valuable approach which supports the accurate detection of nutrients which are deficient phosphorous & Calcium in coffee and Magnesium & Boron in banana.
- By using this project website farmers can identify and recognize the lack phosphorous & Calcium in coffee and Magnesium & Boron in banana in real time and also helps to reduce wait time for the decision of the experts.
- It replaces manual system to digital system and this project approach is User friendly.

CHAPTER 2

METHODOLOGY

DATA COLLECTION

The propose system used Mobile camera to collect the images which is training datasets. The datasets were collected from Coffee and Banana farms. For Coffee Calcium and Phosphorous nutrient deficiency images has been taken to trained coffee model and for Banana Boron and Magnesium deficiency has been taken to trained banana model.

The datasets had been taken as followed:

Magnesium Deficiency Symptoms: Magnesium deficiency symptoms show green in bandingaround the margin and next to the midrib.

Fig: 2.1 Magnesium Deficiency in Banana

- Leaves turn yellowish with brown goods on the leaf margin.
- Plant height reduced marginal yellowing of leaf margin extends towards the midrib.
- Purplish mottling of leaf petiole and malformation of leaves. Fruits do not ripen well and become tasteless.

Boron Deficiency Symptoms: Newly emerging leaves are malformed. Plants sho w shunted growth. Chlorotic streaks appear perpendicular to the veins.

Fig: 2.2 Boron Deficiency in Banana

- Incomplete leaf formation and inhibition of fruit and flower.
- Deficiency of boron may result in reduction in weight and size of the bunchand it will effects the proper filling of the bunch.

Phosphorous Deficiency Symptoms: Reduce growth of plant, restricted development root and leaves dark green.

Fig: 2.3 Phosphorous Deficiency in Coffee

- Symptoms appear in older leaves.
- Irregular yellow areas appear first which may spread to the whole leaf producing mottled appearance.
- Leaves may turn red or violet under severe conditions and drop easily.

Calcium Deficiency Symptoms: Inhibition of bud growth

Fig: 2.4 Calcium Deficiency in Coffee

- leaf tips may stick together; curling/cupping of young leaves.
- Stem structure weakened; premature shedding of fruit and buds.

SYSTEM METHODOLOGY

The main Aim of this project is to help the coffee and banana farmers to classify and recognize the nutrient deficiency in coffee and banana. The project proposed website to classified the lack nutrient in coffee and banana. The website can directly access by coffee and banana farmers. The coffee and banana farmers need to captured the photo and upload to the website then the result willbe display in same page which is about lack nutrient and correct measures. The system uses the image processing technique to trained model, Image processing is a method to perform some operations on an image, in order to get an enhanced image or to extract some useful information from it. Image processing is done by several process such as Collecting the training datasets which is sets of images, Pre-processing data & convertingthe image data to array by OpenCV, Stored the array to Pickle module and Trained the model by CNN algorithm.

- ❑ Compared to Regular Neural Networks, Convolutional Neural Networks (CNN) have various architecture. In Regular Neural Networks, it transforms an input by having multiple series of hidden layers.
- ❑ CNN consists of four layers: Convolution Layer, Activation Layer, Pooling Layer and Fully Connected Layer.

- CNN considers Feature Extraction or Trained Model.

Trained Model is done under Convolution Layer, Activation Layer, Pooling Layer and Fully Connected Layer as shown in figure 2.5.

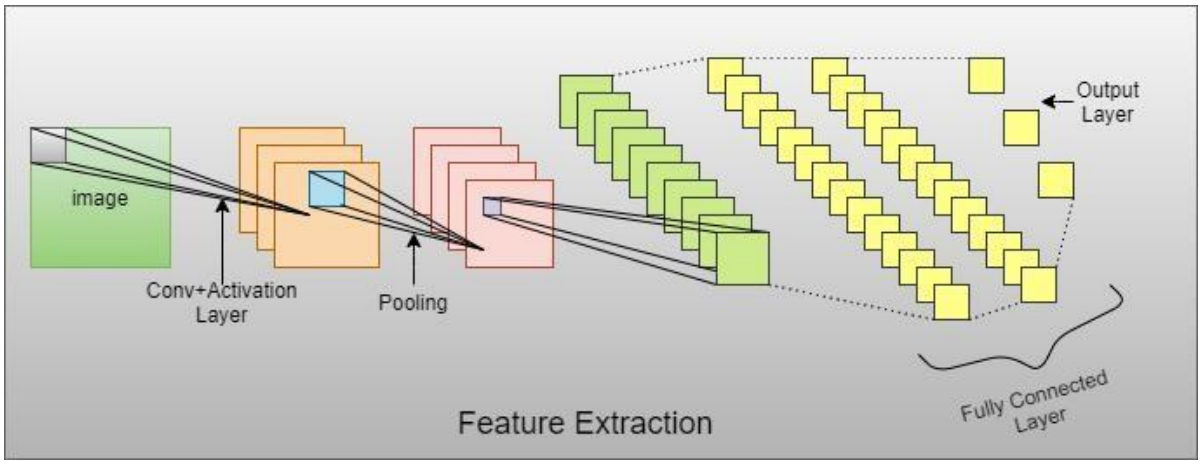

Fig: 2.5 Trained model by CNN algorithm.

Then the trained model is stored with the help of tensorflow.keras.model.

The capture image which is upload to the website by coffee and banana farmers is takenas testing data.

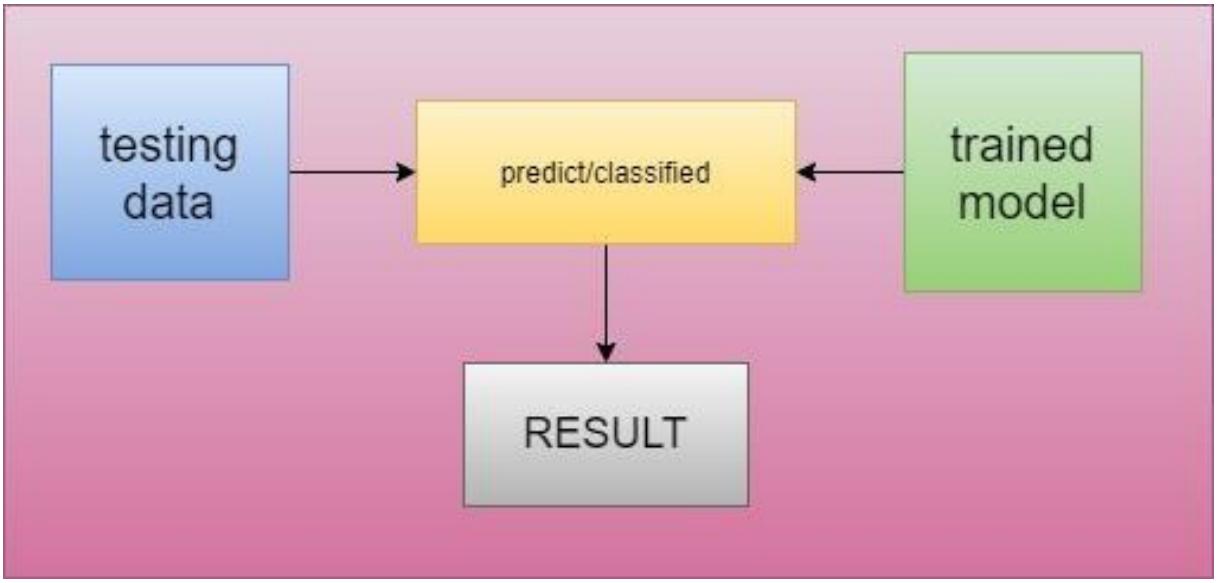

Fig: 2.6 Image Detection.

The testing data will be converted into array and preprocess the model by OpenCV. Then the preprocess model will predict with the train model and gives the result to the website.

CHAPTER 3

SYSTEM DESIGN

LOGICAL DESIGN

Logical design pertains to an abstract representation of the data flow, inputs, and outputs of the system. It describes the inputs (sources), outputs (destinations), databases (data stores), procedures (data flows) all in a format that meets the user requirements. While preparing the logical design of a system, the system analyst specifies the user needs at level of detail that virtually determines the information flow into and out of the system and the required data sources. The project website is web page, the farmers can directly access the web page. The farmer needs to choose and upload the photo. And the result will provide in the same page that the farmer is uploaded.

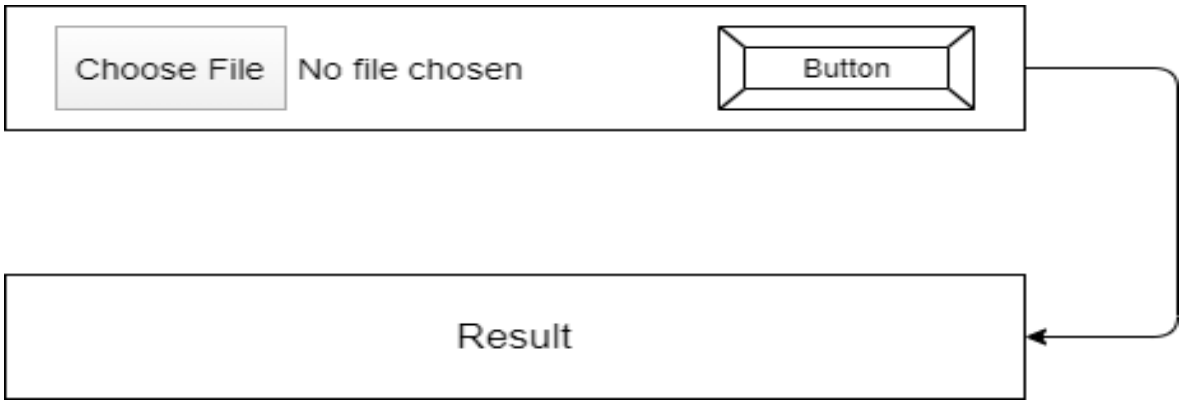

Fig3.1: Logical Design

DESIGN GOALS

The following are the design goals that are applicable to virtually every web app regardless of application domain, size, or complexity.

- Simplicity
- Consistency
- Identity
- Visual appeal
- Compatibility.

Design leads to a model that contains the appropriate mix of aesthetics, content, and web app, and as a consequence the design activities that are emphasized will also vary.

SYSTEM ARCHITECTURE

The upload Image from the website by the farmer is taken as the testing data. The testing data is converted into array and preprocessed then, compare testing data and trained model and gives Result

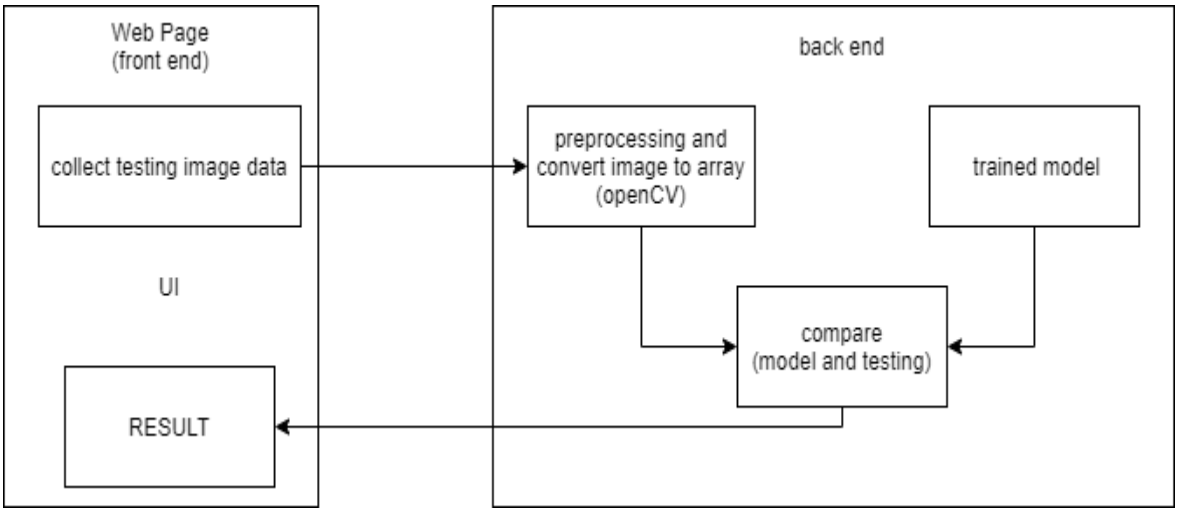

Fig:3.2 System Architecture

The image will be uploaded to the Website User Interface. The uploaded image will be taken as testing data. The testing data which is image data is converted into array and preprocessed by OpenCV. Then with trained model the testing data will predict and gives the result to the Website User Interface.

OpenCV (OpenSource Computer Vision Library) is an open-source library that includes several hundreds of computer vision algorithms. OpenCV is the leading opensource library for computer vision, image processing and machine learning, and now features GPU acceleration for real-time operation. OpenCV 2.x API, which is essentially a C++ API, as opposed to the C- based OpenCV 1.x API.

OpenCV can directly install to anaconda environment by various steps as follows:

1. **Creating Anaconda Environment :**

   Step 1: Search Anaconda in your task bar and select ANACONDA NAVIGATOR.

   Step 2: Now you will see a menu with various options like Jupiter notebook , Spyder etc. This is Anaconda Environment.

   Step 3: Select Jupyter Notebook as it is Anaconda's IDE for python and OpenCV library will work in it.

2. Install OpenCV

Step 1: After installing the anaconda open the Anaconda Prompt.

Step 2: Type the given command, press enter and let it download the whole package.

Command

"pip install opencv-python"

Step 3: Now simply import OpenCV in your python program in which you want to use image processing functions.

Working of OpenCV

- In computers everything videos, documents, images etc. are all converted and stored in form of numbers.
- Pixel value convert images into numbers.
- A pixel is the smallest unit of a digital image.
- The picture intensity at the particular location is represented by the numbers.

OpenCV works in BGR format (blue, green, red)

This stage has two key activities: Understanding or analysis of problem and requirement specification. In problem analysis, the purpose of understanding of the problem and its context as well as the requirements for the new system. Innovative thinking is required to understand the requirements which do not exist for a system. Once Identifying Images, Images are represented in black and white only (white as 255 and black as 0).

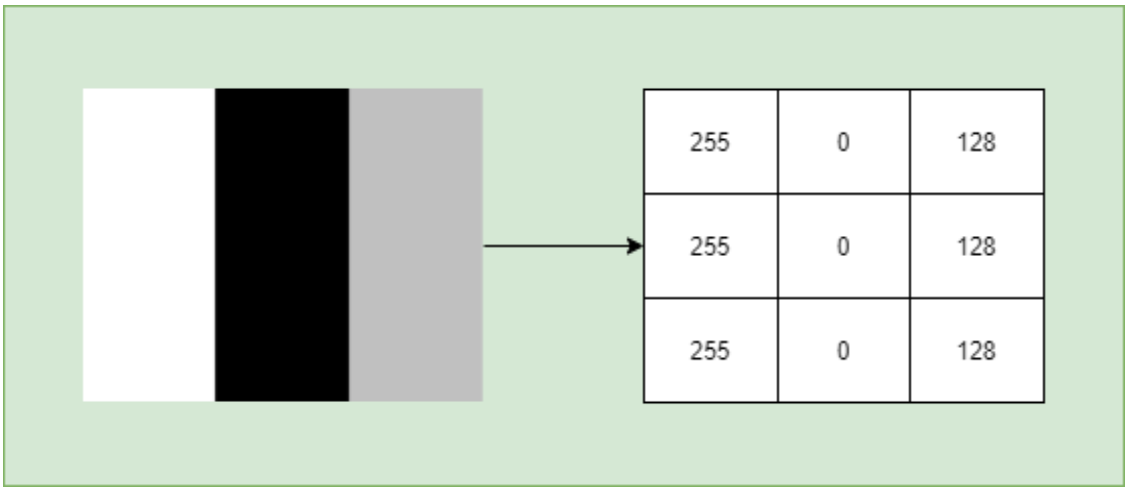

Fig:3.3.1 Image to array conversion.

Images can be classified as:

1. **Grey Scale image**

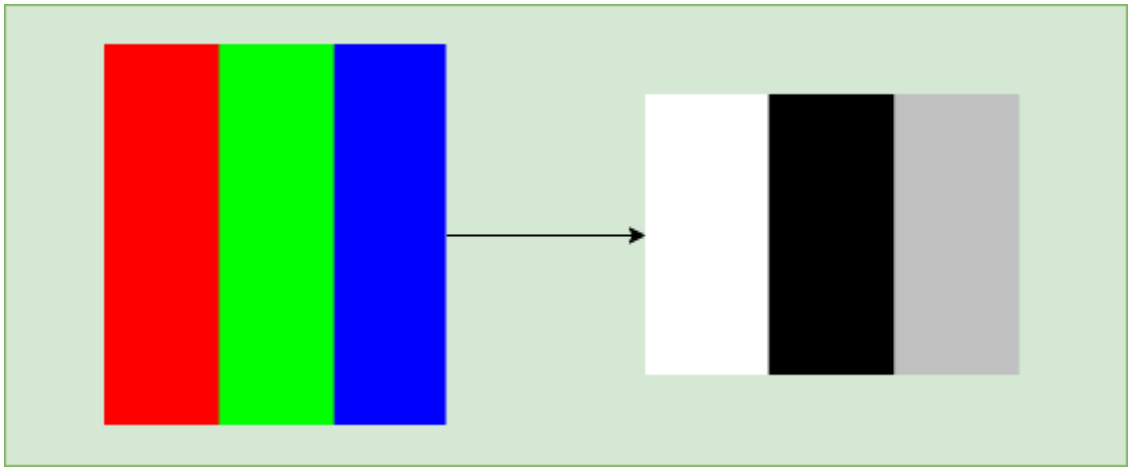

Fig:3.3.2 RBG to Grey Scale image conversion

2. BGR format

Images are combination of three colors blue, green and red. The computer extracts that value from each pixel and puts the results in an array to be interpreted. Images are represented as three channels blue, green and the analysis on the problem and understanding of essentials are done, requirements specification should happen in requirement specification document.Complete behavior detail of a system is described in Software Requirements Specification (SRS). SRS is the goal of the requirements activity. A set of use cases are included which describes the interactions between user and the system. Use cases are

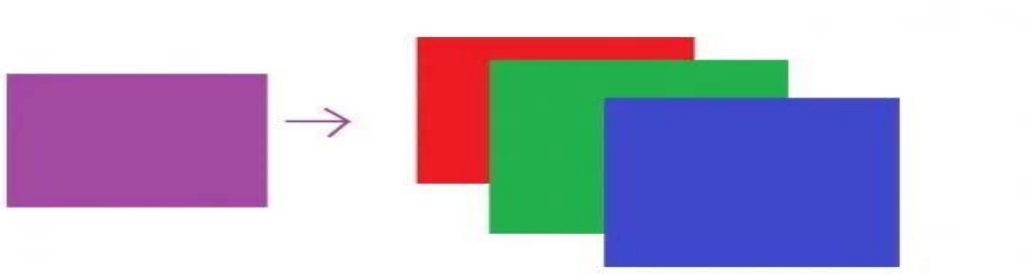

Fig: 3.3.3 Color image to RBG conversion

Application of OpenCV

- Face Detection and Face Recognition are very common applications of OpenCV.
- Self Driving cars also gaining popularity among autonomous Industry.
- Obstacle Avoidance is used for purpose like beach cleaning, home cleaning and many more.

- In Robotics field it is used for localization, navigation, Assembly, human interaction.
- In medical field, it is used for classification and detection problems and building vision guiding robots.
- In Industries also it is used for automation, object sorting, documentation and many more.
- OpenCV is also used for security purpose like catching speedy vehicles.

TRAINING MODEL DESIGN

To trained the model we collected the image data of Banana leaves with Boron & Magnesium Deficiency and Coffee leaves with Calcium and Phosphorous Deficiency. Those images were captured by mobile camera from various Banana and Coffee farms. Then we used those images as training datasets.

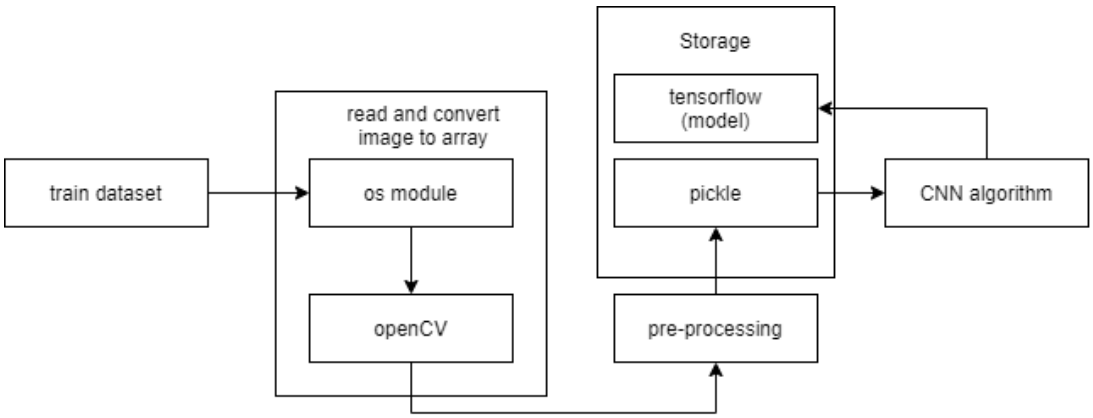

Fig:3.4 Training model

The training datasets were trained as following:

- By OS module the training datasets were called to trained the model. The OS module in Python provides functions for interacting with the operating system. OS comes under Python's standard utility modules. This module provides a portable way of using operating system dependent functionality. The "os" and "os.path" modules include many functions to interact with the file system. It is possible to automatically perform many operating system tasks. The OS module provides functions for creating and removing a directory, fetching its contents, changing and identifying the current directory, etc. You first need to import the os module to interact with the underlying operating system. So, import it using the import os statement before using its functions. This means that it is with

Python installation, but you still must import it. All of the following code assumes you have os imported. Because it is not a built-in function, you must always import it.An operating system is the most important software that runs on a computer. It manages the computer's memory and processes, as well as all of its software and hardware. It also allows you to communicate with the computer without knowing how to speak the computer's language.The training datasets which is read by OS module were converted into array and preprocessed by OpenCV.

- The preprocessed datasets are stored by using pickle module. Pickle module implements binary protocols for serializing and de-serializing a Python object structure. Any object in Python can be pickled so that it can be saved on disk. “Pickling” is the process whereby a Python object hierarchy is converted into a byte stream, and “unpickling” is the inverse operation, whereby a byte stream (from a binary file or bytes-like object) is converted back into an object hierarchy. Pickling (and unpickling) is alternatively known as “serialization”, “marshalling,” 1 or “flattening”; however, to avoid confusion, the terms used here are “pickling” and “unpickling”.

Working of Pickle module

- Pickle in Python is primarily used in serializing and deserializing a Python object structure.
- When a byte stream is unpickled, the pickle module creates an instance of the original object first and then populates the instance with the correct data. To achieve this, the byte stream contains only the data specific to the original object instance. But having just the data alone may not be sufficient. To successfully unpickle the object, the pickled byte stream contains instructions to the unpickle to reconstruct the original object structure along with instruction operands, which help in populating the object structure.
- According to the pickle module documentation, the following types can be pickled:None, true, and false. Integers, long integers, floating point numbers, complex numbersNormal and Unicode stringsTuples, lists, sets, and dictionaries containing only picklable objects. Functions defined at the top level of a moduleBuilt-in functions defined at the top level of a module.

- Pickle allows different objects to declare how they should be pickled using the __reduce__ method. Whenever an object is pickled, the __reduce__ method defined by it gets called. This method returns either a string, which may represent the name of a Python global, or a tuple describing how to reconstruct this object when unpickling.
- Generally the tuple consists of two arguments:
  - A callable (which in most cases would be the name of the class to call)
  - Arguments to be passed to the above callable
- The pickle library will pickle each component of the tuple separately, and will call the callable on the provided arguments to construct the new object during the process of unpickling.
- Pickling allows you to save a python object as a binary file on your hard drive. After you pickle your object, you can kill your python session, reboot your computer if you want, and later load your object into python again.

Convolutional Neural Networks (CNN) algorithm the training datasets were trained by calling pickle module.

Convolutional Neural Networks**,** which is commonly referred to as CNN or ConvNet. Convolutional neural networks have been one of the most influential innovations in the field of computer vision. They have performed a lot better than traditional computer vision and have produced state-of-the-art results. These neural networks have proven to be successful in many different real-life case studies and applications, like: Image classification, object detection, segmentation, face recognition. Self **d**riving cars that leverage CNN based vision systems. The CNN which are the convolutional layers, pooling layers, and fully-connected (FC) layers. When these layers are stacked, a CNN architecture will be formed. Three layers and there are two more important parameters which are the dropout layer and the activation function

Feature extraction by using Convolutional Neural Networks.

There are various types of layers that make up the CNN which are the convolutional layers, pooling layers, and fully-connected (FC) layers. When these layers are stacked, a CNN architecture will be formed. Three layers and there are two more important parameters which are the dropout layer and the activation function as follows:

1. Convolutional Layer

- This layer is the first layer that is used to extract the various features from the input images. In this layer, the mathematical operation of convolution is performed between the input image and a filter of a particular size MxM. By sliding the filter over the input image, the dot product is taken between the filter and the parts of the input image with respect to the size of the filter (MxM).
- The output is termed as the Feature map which gives us information about the image such as the corners and edges. Later, this feature map is fed to other layers to learn several other features of the input image.

2. Pooling Layer

- In most cases, a Convolutional Layer is followed by a Pooling Layer. The primary aim of this layer is to decrease the size of the convolved feature map to reduce the computational costs. This is performed by decreasing the connections between layers and independently operates on each feature map. Depending upon method used, there are several types of Pooling operations.
- In Max Pooling, the largest element is taken from feature map. Average Pooling calculates the average of the elements in a predefined sized Image section. The total sum of the elements in the predefined section is computed in Sum Pooling. The Pooling Layer usually serves as a bridge between the Convolutional Layer and the FC Layer

3. Fully Connected Layer

- The Fully Connected (FC) layer consists of the weights and biases along with the neurons and is used to connect the neurons between two different layers. These layers are usually placed before the output layer and form the last few layers of a CNN Architecture.
- In this, the input image from the previous layers are flattened and fed to the FC layer. The flattened vector then undergoes few more FC layers where the mathematical functions operations usually take place. In this stage, the classification process begins to take place.

4. Dropout

- Usually, when all the features are connected to the FC layer, it can cause overfitting in the training dataset. Overfitting occurs when a particular model works so well on the training data causing a negative impact in the model's performance in various form while using on already existing data or new data.

- To overcome this problem, a dropout layer is utilized wherein a few neurons are dropped from the neural network during training process resulting in reduced size of the model. On passing a dropout of 0.3, 30% of the nodes are dropped out randomly from the neural network.

5. Activation Functions

Finally, one of the most important parameters of the CNN model is the activation function. They are used to learn and approximate any kind of continuous and complex relationship between variables of the network. In simple words, it decides which information of the model should fire in the forward direction and which ones should not at the end of the network.

Store the trained model to tensorflow.keras.model

TensorFlow.keras.model, is an open-source software library that provides a Python interface for Neural Networks. Tensorflow.keras acts as an interface for the TensorFlow library. It was developed with a focus on enabling fast experimentation. Tensorflow.keras reduces developer cognitive load to free you to focus on the parts of the problem that really matter. Tensorflow.keras is a high-level interface and uses Theano or Tensorflow for its backend. It runs smoothly on both CPU and GPU. Tensorflow.keras supports almost all the models of a neural network – fully connected, convolutional, pooling, recurrent, embedding, etc. Furthermore, these models can be combined to build more complex models.The key idea behind the development of Tensorflow.keras is to facilitate experimentations by fast prototyping.

Tensorflow.keras Application:

- **Simple** -- but not simplistic. Tensorflow.keras reduces developer cognitive load to free you to focus on the parts of the problem that really matter.
- **Flexible** -- Tensorflow.keras adopts the principle of progressive disclosure ofcomplexity: simple workflows should be quick and easy, while arbitrarily advanced workflows should be possible via a clear path that builds.
- **Powerful** -- Tensorflow.keras provides industry-strength performance and
- **Scalability:** it is used by organizations and companies including NASA, YouTube, or Waymo.

TensorFlow 2 is an end-to-end, open-source machine learning platform. It combines four key abilitiesEfficiently executing low-level tensor operations on CPU, GPU, or TPU.Computing the gradient of arbitrary differentiable expressions.Scaling computation to many devices, such as clusters of hundreds of GPUs.Exporting programs ("graphs") to external runtimes such as servers, browsers, mobile and embedded devices.Tensorflow.keras is the high-level API of TensorFlow 2: an approachable, highly-productive interface for solving machine learning problems, with a focus on modern deep learning. It provides essential abstractions and building blocks for developing and shipping machine learning solutions with high iteration velocity.Tensorflow.keras empowers engineers and researchers to take full advantage of the scalability and cross-platform capabilities of TensorFlow 2: you can run Tensorflow.keras on TPU or on large clusters of GPUs, and you can export your Tensorflow.keras models to run in the browser or on a mobile device.The core data structures of Tensorflow.keras are layers and models. The simplest type of model is the Sequential model, a linear stack of layers. For more complex architectures, you should use the Tensorflow.keras functional API, which allows to build arbitrary graphs of layers, or write models entirely from scratch via subclass.

PREDICTION PROCESS DESIGN

The trained model, which is stored in tensorflow.keras.model will be called and comparing with the testing data by using keras.model.predict.

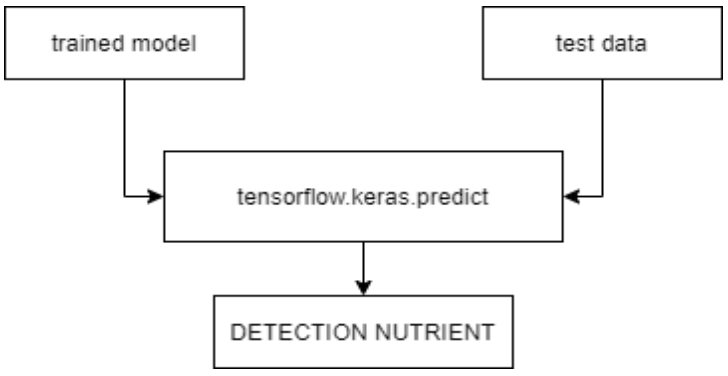

Fig: 3.5 Prediction process

Tensorflow.keras models can be used to detect trends and make predictions, using the *model.predict()* class**.** A model can be created and fitted with trained data, and used to make a prediction.

SOFTWARE REQUIREMENT SPECIFICATION

The purpose of this document is to build a Website to detect the lack nutrients in Coffee and Banana. The Website will help the Coffee and Banana Farmer to detect the lack nutrients. The Website can detect lack nutrient such as Boron & Magnesium in Banana and Calcium & Phosphorous in Coffee. The Website can be accessed by the farmers directly.

Specific Requirements

To detect the lack nutrient, first of all the training datasets which is image datasets. The training image datasets are collected by capturing images using mobile camera. And for upload the image by farmers to the website, they also need to be capture using any of camera, image should be color image only.

Functional Requirements

- Information about Correction measure of the lack nutrients such as Calcium & Phosphorous for Coffee and Boron & Magnesium for Banana.
- Python, Html and CSS language.
- Django web framework.
- URL to link between python and html.
- Training Datasets.

Non-Functional Requirements

- Performance: The time complexity depends on speed of internet.
- Data Integrity: While training the model Banana predicted 87.60% accuracy and Coffee predicted 75.71 % accuracy.
- Maintainability: The web page is easy to used and it is user friendly.
- Reliability: The system can measure accurate lack nutrients such as Calcium & Phosphorous for Coffee and Boron & Magnesium for Banana.
- Usability: The project website can be Directly accessed.

Software and Hardware Requirements

Following are the software and hardware requirements in order to carry out the project:

- System : Desktop/Laptop
- Operating System : Windows 7/8/10
- Processor : Platinum IV onwards
- Hard Disk : 100 GB
- Ram : 4 GB

- **To Trained the model:**

- Tools : Jupyter Notebook

- **To Create web page:**

- Tools : Visual Studio Code
- Browser : Any browser

- **For User:**

- Good Internet Connection
- System : Mobile
- Operating System : Android/iOS
- Browser : Any browser

CHAPTER 4

IMPLEMENTATION

OVERVIEW

An implementation is a realization of a technical specification or algorithm as a program, software component or other computer system through computer programming and deployment. Many implementations may exists for a given specification or standard. To implement a system successfully, a large number of inter-related tasks need to be carried out in an appropriate sequence.

Implementation is the stage in the project where theoretical design is turned into a working system and is giving confidence on the new system for the users that it will work efficiently and effectively. It involves careful planning, investigation of the current system and its constraints on implementation, design of methods to achieve the changeover, an evaluation of change over methods. The implementation process begins with the preparing a plans for the implementation of the system. According to this plan, the activities are to be carried out, discussions made regarding the equipment and resources and the additional equipment has to be acquired to implement the new system.

Implementation is the most important and critical phase in achieving a successful new system, that gives the user confidence that the new system will work and is effective. The system can be implemented only after through testing is done and if it is found to be working according to the specification.

The model is trained and stored by **Python** language using **CNN** algorithm in **Jupyter Notebook**. And the front end and the back end of web page is construct by using **Django** web framework.

PYTHON

Python is a widely used general-purpose, high level programming language. It was designedbyGuidovanRossumin1991anddevelopedbyPythonSoftwareFoundation.It was mainly developed for emphasis on code readability, and its syntax allows programmers to express concepts in fewer lines of code. Python is a high-level language, meaning that it abstracts underlying computer-related technical details. For example, Python does not make its users think too much about computer memory management or proper declaration of variables and uses safe assumptions about what the programmer is trying to convey. In addition, a high-level language can be expressed in a manner closer to

English prose or mathematical equations. Python is perfect for literate programming because of its lightweight, “low ceremony” nature. Python is a general-purpose specializing in a specific area such as statistical analysis. For example, Python can be used for both artificial intelligence and statistical analysis. Python can be used for a variety of heterogeneous tasks within a given work-flow, as the UCAR scientist described earlier. Python is an interpreted language meaning that evaluation of code to obtain resultscan happen immediately rather than having to go through a time-consuming, compile and run cycle, which thereby speeds up the thinking and experimentation processes. Python has a standard library, and numerous third-party libraries yielding a vast array of existing codebases and examples for solving problems.

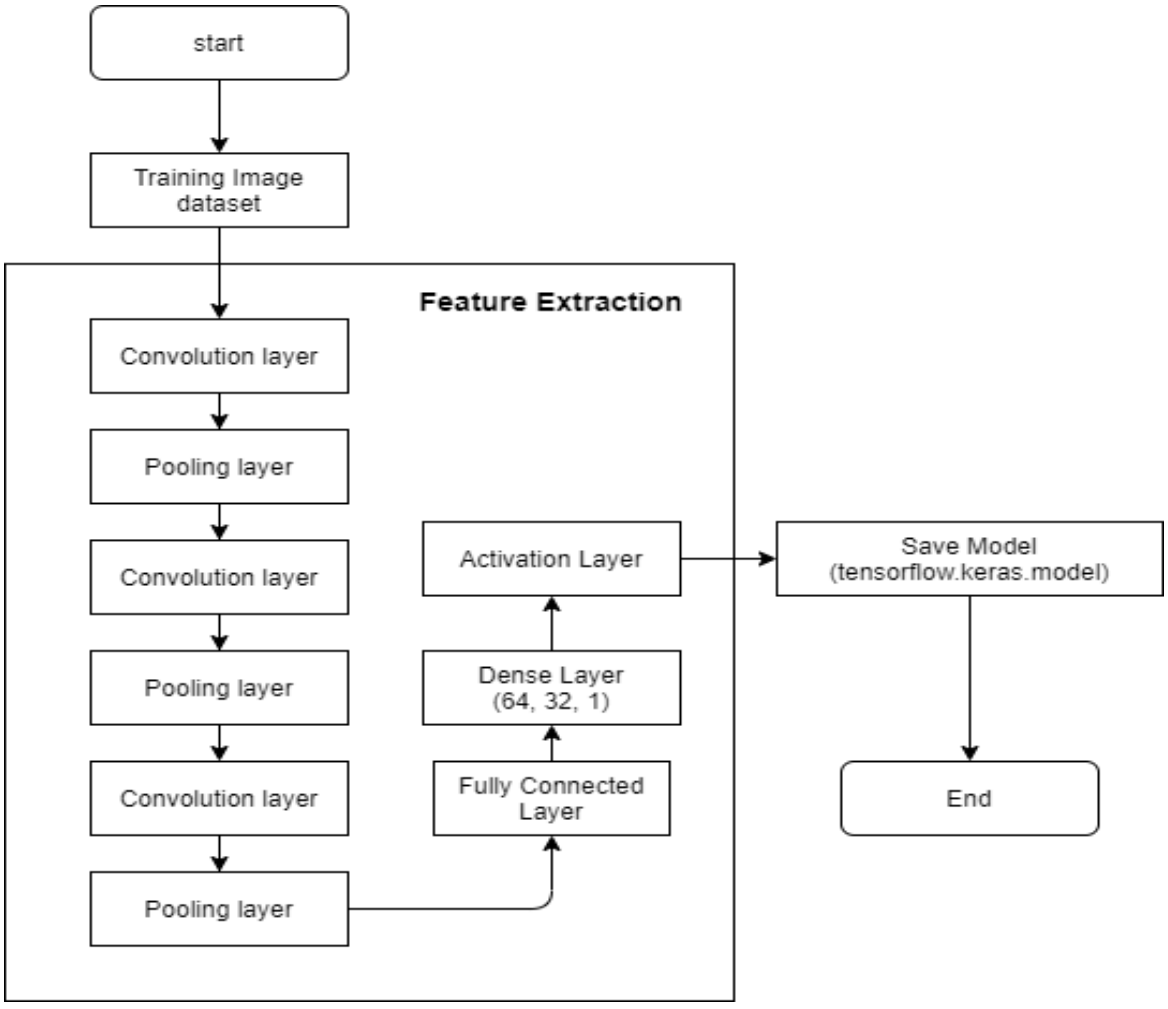

Fig:4.1 Flowchart of implement CNN algorithm.

Jupyter Notebook

Jupyter Notebook (formerly IPython Notebooks) is a web-based interactive computational environment for creating Jupyter notebook documents.

The "notebook" term can colloquially make reference to many different entities, mainly

the Jupyter web application, Jupyter Python web server, or Jupyter document format depending on context. A Jupyter Notebook document is a JSON document, following a versioned schema, containing an ordered list of input/output cells which can contain code, text (using Markdown), mathematics, plots and rich media, usually ending with the ".ipynb" extension.AJupyter Notebook can be converted to a number of open standard output formats (HTML, presentation slides, LaTeX, PDF, ReStructuredText, Markdown, Python) through "Download As" in the web interface, via the nbconvert library[9] or "jupyternbconvert" command line interface in a shell. To simplify visualisation of Jupyter notebook documents on the web, the nbconvertlibrary[10] is provided as a service through NbViewer[11] which can take a URL to any publicly available notebook document, convert it to HTML on the fly and display it to the user.

The Jupyter Notebook is an open-source web application that allows you to create and share documents that contain live code, equations, visualizations and narrative text. Uses include: data cleaning and transformation, numerical simulation, statistical modeling, data visualization, machine learning, and much more.TheJupyter Notebook App is a server-client application that allows editing and running notebook documents via a web browser. The Jupyter Notebook App can be executed on a local desktop requiring no internet access (as described in this document) or can be installed on a remote server and accessed through the internet.

```
for category in CATEGORIES:
    path = os.path.join(DATADIR,category)
    for img in os.listdir(path):
        img_array = cv2.imread(os.path.join(path,img))
        cv2.imread(os.path.join(path,img))

        break
    break

def create_training_data():
    for category in CATEGORIES:

        path = os.path.join(DATADIR,category)
        class_num = CATEGORIES.index(category)

        for img in tqdm(os.listdir(path)):
            try:
                img_array = cv2.imread(os.path.join(path,img))
                new_array = cv2.resize(img_array, (IMG_SIZE, IMG_SIZE))
                training_data.append([new_array, class_num])
            except Exception as e:
                pass
```

```
import pickle

pickle_out = open("C1.pickle","wb")
pickle.dump(C1, pickle_out)
pickle_out.close()

pickle_out = open("C2.pickle","wb")
pickle.dump(C2, pickle_out)
pickle_out.close()
pickle_in = open("C1.pickle","rb")
C1 = pickle.load(pickle_in)
pickle_in = open("C2.pickle","rb")
C2 = pickle.load(pickle_in)

model = Sequential()
model.add(Conv2D(256, (3, 3), input_shape=x.shape[1:]))
#model.add(Activation('sigmoid'))
model.add(MaxPooling2D(pool_size=(2, 2)))

model.add(Conv2D(256, (3, 3)))
#model.add(Activation('sigmoid'))
model.add(MaxPooling2D(pool_size=(2, 2)))

model.add(Conv2D(256, (3, 3)))
model.add(MaxPooling2D(pool_size=(2, 2)))

model.add(Flatten())
model.add(Dense(64))
model.add(Dense(32))
model.add(Dense(1))
model.add(Activation('sigmoid'))

model.compile(loss='binary_crossentropy',
              optimizer='adam',
              metrics=['accuracy'])

model.fit(x, y, batch_size=32, epochs=4, validation_split=0.1)

model.save("trained_model.model")
```

Fig4.2.1: Python code to trained model

DJANGO

Django is an advanced Web framework written in Python that makes use of the model view controller (MVC) architectural pattern. Django was created in a fast-moving newsroom environment, and its key objective is to ease the development of complicated, database-driven websites. This Web framework was initially developed for The World Company for managing some of their news-oriented sites. In July 2005, it was publicly released under a BSD license. Built by experienced developers, Django takes care of much of the hassle of web development. It is free and , has a thriving and active community, great documentation, and many options for free and paid-for support. Django is available as an open-source Web framework, and it uses Python extensively to create files, settings and data models. Django can be used to build almost any type of website — from content management systems and wikis, through to social networks and news sites. It can work with any client-side framework, and can deliver content in almost any format including HTML, RSS feeds, JSON, XML, etc. Django helps developers avoid many common security mistakes by providing a framework that has been engineered to "do the right things" to protect the website automatically.

Visual Studio Code

Visual Studio Code is an integrated development environment (IDE) from Microsoft. It is used to develop computer programs, as well as websites, web apps, web services and mobile apps. Visual Studio uses Microsoft software development platforms such as Windows API, Windows Forms, Windows Presentation Foundation, Windows Store and Microsoft Silverlight. It can produce both native code and managed code.Visual Studio includes a code editor supporting IntelliSense (the code completion component) as well as code refactoring. The integrated debugger works both as a source-level debugger and a machine-level debugger. Other built-in tools include a code profiler, designer for building GUI applications, web designer, class designer, and database schema designer. It accepts plug-ins that expand the functionality at almost every level—including adding support for source control systems (like Subversion and Git) and adding new toolsets like editors and visual designers for domain-specific languages or toolsets for other aspects of the software development lifecycle (like the Azure DevOps client: Team Explorer).

Visual Studio supports 36 different programming languages and allows the code editor and debugger to support (to varying degrees) nearly any programming language, provided a language-specific service exists. Built-in languages include C,C++, C++/CLI,

Visual Basic .NET, C#, F#,JavaScript, TypeScript, XML, XSLT, HTML, and CSS. Support for other languages such as Python, Ruby, Node.js, and M among others is available via plug-ins. Java (and J#) were supported in the past.The most basic edition of Visual Studio, the Community edition, is available free of charge. The slogan for Visual Studio Community edition is "Free, fully-featured IDE for students, open-source and individual developers".

HTML

The HyperText Markup Language, or HTML is the standard markup language for documents designed to be displayed in a web browser. It can be assisted by technologies such as Cascading Style Sheets and scripting languages such as JavaScript.

```
<body>

{% load static %}
<button type="button" class="collapsible">About us</button>

<div class="contain" style="background: #008000;">
      <h1><center>Detection lack of Nutrient in Coffee and Banana by Image
 Processing</center></h1>
</div>

<div class="container"  style="background-
image: url('{%static '/banana.jpg'%}');">
<center>
<form action="banana" method="post" enctype="multipart/form-data" >
{% csrf_token %}
<br><br>
<input type="file" name="image" required>
<input type="submit" value="Click Here for Banana" >
</center>
<div>
  <div><h1>
    {% if data %}
    {{data}}
    {% endif %}
  </h1>
  </div>
  <div><h1>
    {% if Correction_B %}
    {{Correction_B}}
    {% endif %}
```

```
    </h1>
  </div>
</div>
</form>
 </div>

<div class="container" style="background-
image: url('{%static '/coffee.jpg'%}');">
<center>
  <form action="coffee" method="post" enctype="multipart/form-data" >
    {% csrf_token %}
    <br><br>
    <input type="file" name="image" required>
    <input type="submit" value="Click Here for Coffee" >
    <div >
      <div><h1>
        {% if data_C %}
        <span>
        {{data_C}}
        </span>
        {% endif %}
      </h1>
      </div>
      <div>
        <h1>
        {% if Correction_C %}
        {{Correction_C}}

{% endif %}
        </h1>
    </div>
    </form>

</div>
</center>
</div>
</body>
```

Fig 4.2.2: Html code.

CSS

Cascading Style Sheets is a style sheet language used for describing the presentation of a document written in a markup language such as HTML. CSS is a cornerstone technology of the World Wide Web, alongside HTML and JavaScript.

```
.button
  {
    background-color: darkgreen;
    color: white;
    padding: 14px 20px;
    margin: 8px 0;
    width: 350px;
    height: 40px;
    font-size: 15px;
    align-items: right;
  }
  div{
    background-color: darkgreen;
    color: white;
    }
  h1,label
  {
    color: white;
  }

  p
  {
    font-style: italic;
  }
  .container
  {
    padding: 5%;
  }
  .contain
  {
    padding: 0.25%;
  }

  input[type="file"] {
  position: inherit;
  z-index: -1;
  top: 10px;
        left: 8px;
        font-size: 17px;
        color: #b8b8b8;
      }
```

Fig 4.2.3: CSS code.

CHAPTER 5

TESTING

OVERVIEW

Testing is basically a process of technical enquiry i.e. performed on behalf of stakes holders that intends to disclose the quality-related data about the product.

Software testing is an investigation conducted to provide stakeholders with information about the quality of the product or service under test. It involves the execution of system component to evaluate one or more properties of interest. In general, these properties indicate the extent to which the component or system under test:

- ✓ Meets the requirements that guided its design and development,
- ✓ Performs its functions within an acceptable time,
- ✓ Can be accessed in browsers, and
- ✓ Achieves the general result.

TEST CASES:

The test cases in topic and the outcomes that are expected according to test plan developed for Project proposal.

Table 5.1 Test case for opening website correctly.

Name of test	Open website
Simple Input	Run server 127.0.0.1:8000
Expected Output	Web page contain “About us”-button, “choose file”-file, “chick herefor Banana”-button, “choose file”-file, “chick here for Coffee”-button.
Actual Output	Web page contain “About us”-button, “choose file”-file, “chick here for Banana”-button, “choose file”-file, “chick here for Coffee”-button.
Remarks	Website opening Correctly.

Table 5.2 Test case for opening website wrongly.

Name of test	Open website
Simple Input	Run server 127.0.0.1:8000
Expected Output	Web page contain “About us”-button, “choose file”-file, “chick here for Banana”-button, “choose file”-file, “chick here for Coffee”-button.
Actual Output	This site can’t be reach.
Remarks	Website opening wrongly.

Table 5.3 Test case for predict Magnesium deficiency in Banana correctly.

Name of test	Choose File
Simple Input	Mg Deficiency Image
Expected Output	About “Mg Deficiency” and “Correction Measure”.
Actual Output	About “Mg Deficiency” and “Correction Measure”.
Remarks	Correct prediction.

Table 5.4 Test case for predict Magnesium deficiency in Banana wrongly.

Name of test	Choose File
Simple Input	Mg Deficiency Image
Expected Output	About “Mg Deficiency” and “Correction Measure”.
Actual Output	About “Br Deficiency” and “Correction Measure”.
Remarks	Wrong prediction.

Table 5.5 Test case for predict Boron deficiency in Banana correctly.

Name of test	Choose File
Simple Input	Br Deficiency Image
Expected Output	About “Br Deficiency” and “Correction Measure”.
Actual Output	About “Br Deficiency” and “Correction Measure”.
Remarks	Correct prediction.

Table 5.6 Test case for predict Boron deficiency in Banana wrongly.

Name of test	Choose File
Simple Input	Br Deficiency Image
Expected Output	About “Br Deficiency” and “Correction Measure”.
Actual Output	About “Mg Deficiency” and “Correction Measure”.
Remarks	Wrong prediction.

Table 5.7 Test case for predict Phosphorous deficiency in Coffee correctly.

Name of test	Choose File
Simple Input	P Deficiency Image
Expected Output	About “P Deficiency” and “Correction Measure”.
Actual Output	About “P Deficiency” and “Correction Measure”.
Remarks	Correct prediction.

Table 5.8 Test case for predict Phosphorous deficiency in Coffee wrongly.

Name of test	Choose File
Simple Input	P Deficiency Image
Expected Output	About “P Deficiency” and “Correction Measure”.
Actual Output	About “Ca Deficiency” and “Correction Measure”.
Remarks	Wrong prediction.

Table 5.9 Test case for predict Calcium deficiency in Coffee correctly.

Name of test	Choose File
Simple Input	Ca Deficiency Image
Expected Output	About “Ca Deficiency” and “Correction Measure”.
Actual Output	About “Ca Deficiency” and “Correction Measure”.
Remarks	Correct prediction.

Table 5.10 Test case for predict Calcium deficiency in Coffee wrongly.

Name of test	Choose File
Simple Input	Ca Deficiency Image
Expected Output	About “Ca Deficiency” and “Correction Measure”.
Actual Output	About “P Deficiency” and “Correction Measure”.
Remarks	Wrong prediction.

CHAPTER 6

RESULTS

This Section includes all the snapshots and the screenshots, which shows the application and its interfaces.

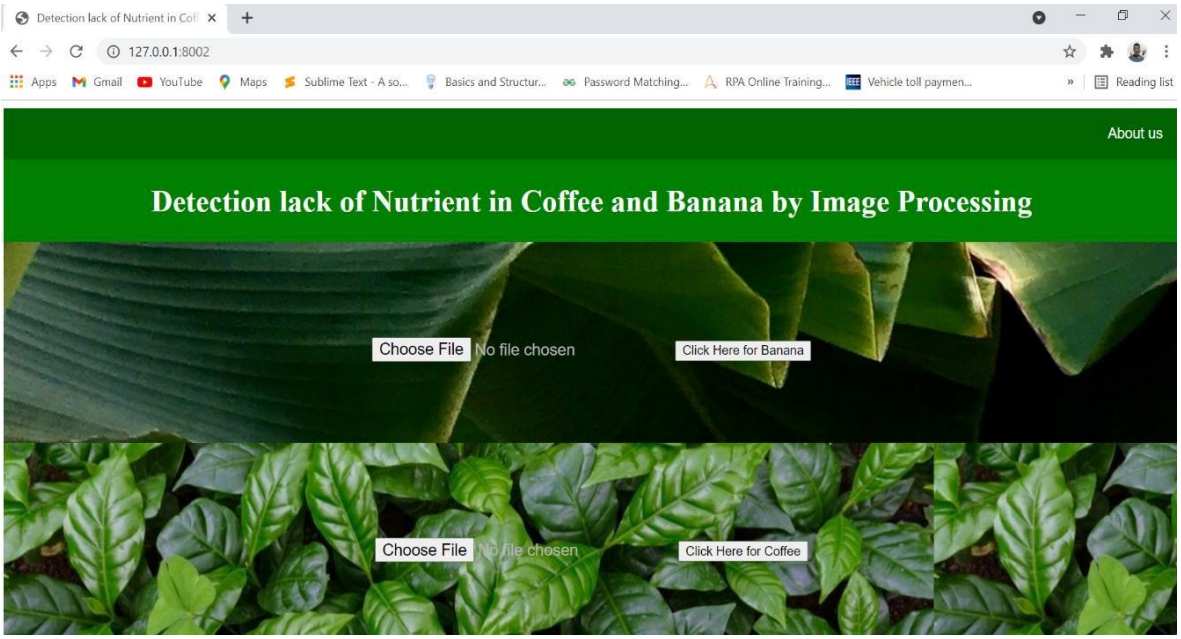

Fig: 6.1 Project web page.

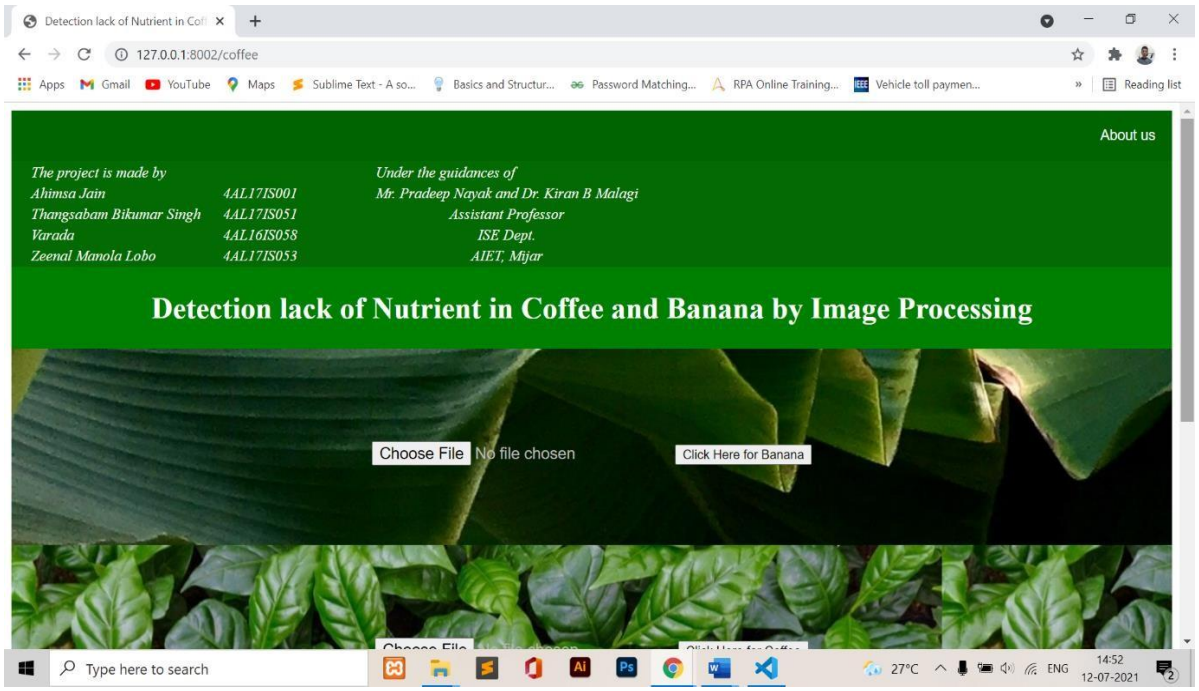

Fig: 6.2 About us.

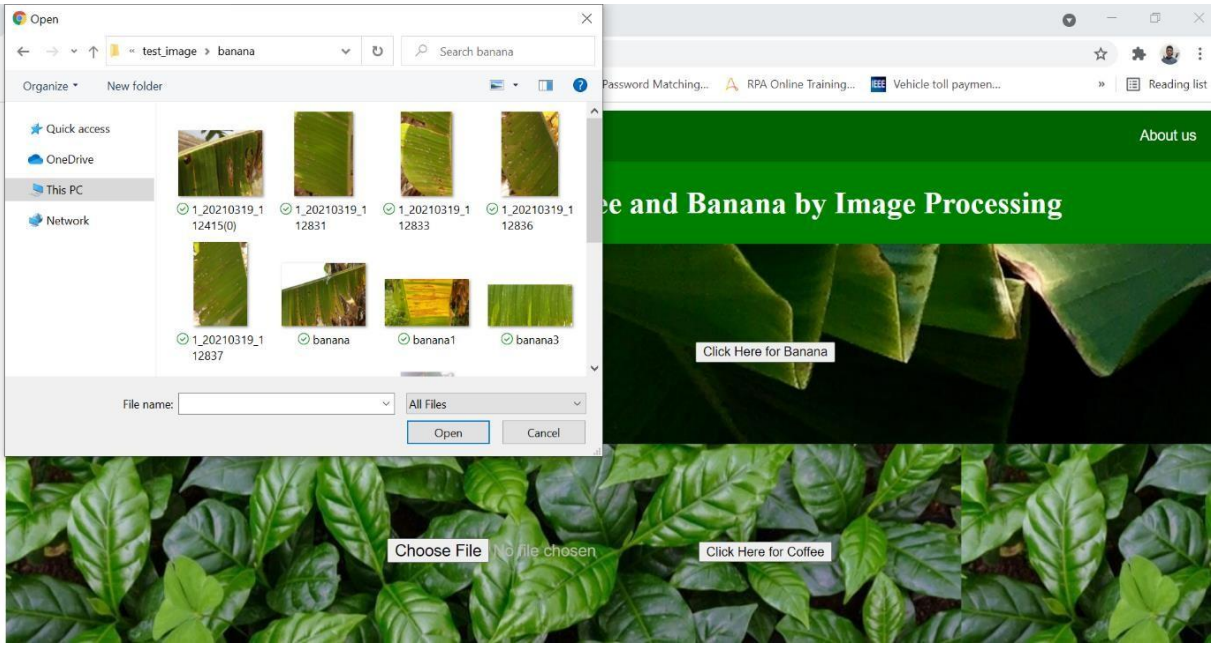

Fig:6.3 Choose file.

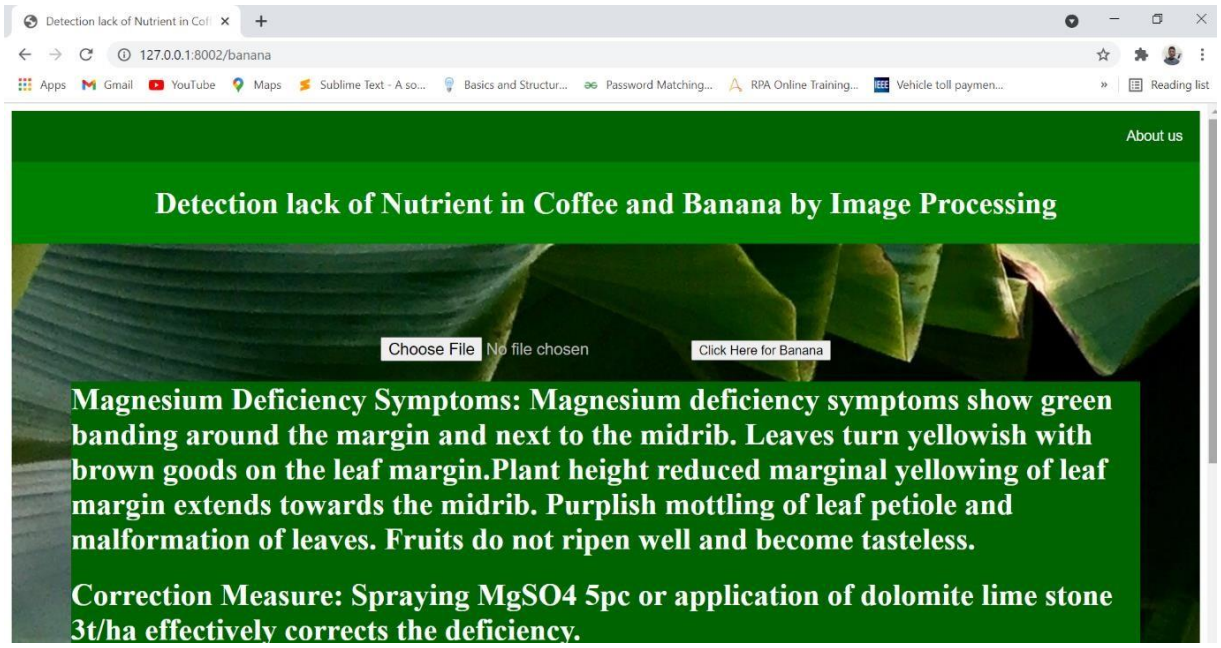

Fig: 6.4 Result for lack nutrient in Banana.

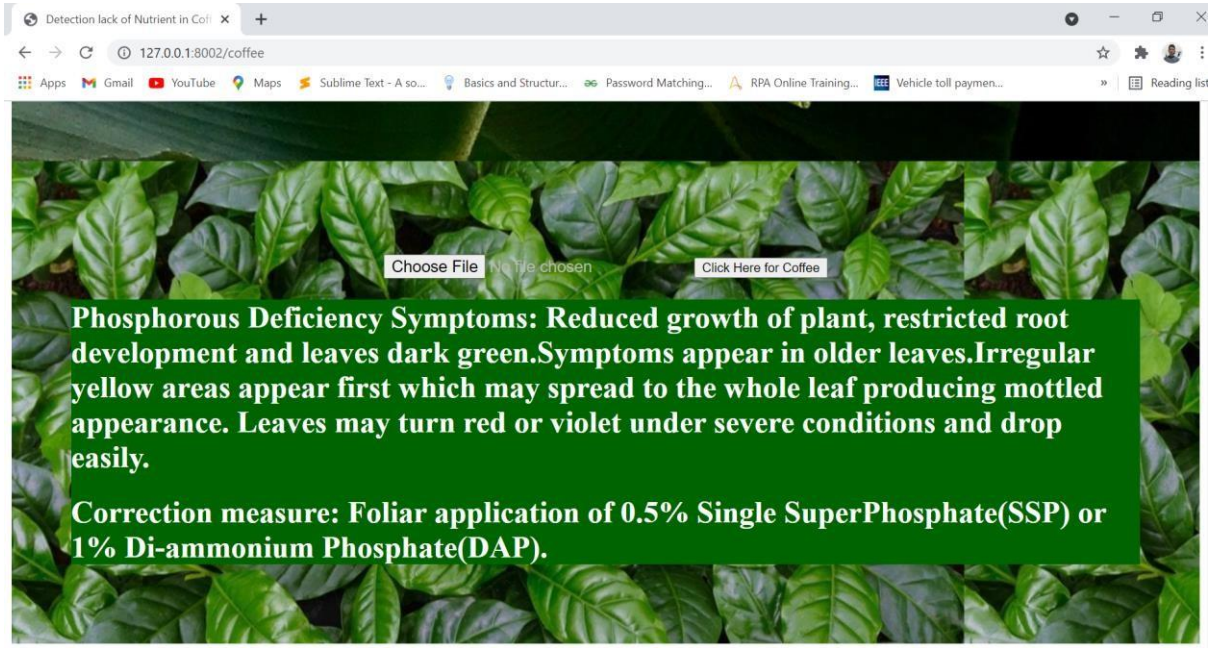

Fig: 6.5 Result for lack nutrient in Coffee

CHAPTER 7

CONCLUSION AND FUTURE ENHANCEMENT

In the developing world, technologies play a vital role in all sectors. Human survival mainly depends on Agriculture. Nowadays the growth of plants, crops and fruits are normally affected by the lack of Nutrients and diseases. Which affects the growth and yields of the plants which causes heavy loss for the farmers to satisfy the need of growing population, it is high time to focus on maximum yield. This is possible only if plants get sufficient nutrient for their growth. The proposed system is based on detection of lack of nutrients deficiency in coffee and banana using Image Processing. This project helps the Coffee and Banana growing farmers to efficiently classify and recognize lack of Calcium & Phosphorous in Coffee and Boron & Magnesium in Banana. The project proposed website, which the farmers can directly access. This project provides the efficient result with high accuracy and efficiency. This project proposes a valuable approach which identifies the accurate nutrients which are Calcium & Phosphorous deficient in Coffee and Boron & Magnesium deficient in Banana. With the help of this project farmers can identify and recognize the lack nutrient deficiencies in real time and also helps to reduce wait time for the decision of the experts. It replaces manual system to digital system and this project approach is User friendly. It would Promote Indian farmers to do smart farming which helps to take time to time decisions is also save time and reduce loss of fruit due to diseases. Farmers usually identify symptoms of disorder on fruit. Expert's takes less time to detect disorder on the fruit but the major problem It is not possible to get the decision on time due. So, in such a situation this system will help the farmers

7.1 FUTURE ENHANCEMENT:

The proposed system will be very useful if it is integrated with many more facilities like fertilizer suggestion, leaf age detection and identifying the natural aging of the leaf.

CHAPTER 7
REFERENCES

[1] Khenilyn P. Lewis, Juancho D. Espineli.” Classification And Detection Of Nutritional Deficiencies In Coffee Plants Using Image Processing And Convolutional Neural Network (CNN)”

[2] Diego Monsalve, Maria Trujillo, Deisy Chaves.”Automatic Classification of Nutritional Deficiencies in Coffee Plants”.

[3] Ukrit Watchareeruetai, Pavit Noinongyao, Chaiwat Wattanapaiboonsuk, Puriwat Khantiviriya.” Identification of Plant Nutrient Deficiencies Using Convolutional Neural Networks”.

[4] Jose Sosa, Janidet Ramirez, Luis Vives and Guillermo Kemper.“An Algorithm For Detection of Nutritional Deficiencies from Digital Images of Coffee Leaves Based on Descriptors and Neural Networks "