A Project Report on

E-fresh : Computer Vision and IOT Framework for Fruit Freshness Detection

Submitted in partial fulfillment of the requirements for the award of the degree of

Bachelor of Engineering

in

Information Technology

by

Siddhesh Gaikwad(18104069)
Krishita Tolia(18104021)
Akshata Gawas(18104039)

Under the Guidance of

Prof. Kiran Deshpande
Prof. Sonal Jain and Prof. Kaushiki Upadhayaya

Department of Information Technology
NBA Accredited
A.P. Shah Institute of Technology
G.B.Road,Kasarvadavli, Thane(W), Mumbai-400615
UNIVERSITY OF MUMBAI

Academic Year 2021-2022

Approval Sheet

This Project Report entitled ***"E-fresh : Computer Vision and IOT Framework for Fruit Freshness Detection"*** Submitted by ***"Siddhesh Gaikwad"(18104069), "Krishita Tolia"(18104021), "Akshata Gawas"(18104039)*** is approved for the partial fulfillment of the requirenment for the award of the degree of ***Bachelor of Engineering*** in ***Information Technology*** from ***University of Mumbai***.

(Prof. Sonal Jain and Prof. Kaushiki Upadhayaya)
Co-Guide

(Prof. Kiran Deshpande)
Guide

Prof. Kiran Deshpande
Head Department of Information Technology

Place:A.P.Shah Institute of Technology, Thane
Date:

CERTIFICATE

This is to certify that the project entitled ***"E-fresh : Computer Vision and IOT Framework for Fruit Freshness Detection"*** submitted by ***"Siddhesh Gaikwad" (18104069), "Krishita Tolia" (18104021), "Akshata Gawas" (18104039)*** for the partial fulfillment of the requirement for award of a degree ***Bachelor of Engineering*** in ***Information Technology***,to the University of Mumbai,is a bonafide work carried out during academic year 2021-2022.

(Prof. Sonal Jain and Prof. Kaushiki Upadhayaya)
Co-Guide

(Prof. Kiran Deshpande)
Guide

Prof. Kiran Deshpande
Head Department of Information Technology

Dr. Uttam D.Kolekar
Principal

External Examiner(s)

1.

2.

Place:A.P.Shah Institute of Technology, Thane
Date:

Acknowledgement

We have great pleasure in presenting the report on **E-fresh : Computer Vision and IOT Framework for Fruit Freshness Detection.** We take this opportunity to express our sincere thanks towards our guide **Prof. Kiran Deshpande** & Co-Guide **Prof. Sonal Jain and Prof. Kaushiki Upadhayaya** Department of IT, APSIT thane for providing the technical guidelines and suggestions regarding line of work. We would like to express our gratitude towards his constant encouragement, support and guidance through the development of project.

We thank **Prof. Kiran B. Deshpande** Head of Department,IT, APSIT for his encouragement during progress meeting and providing guidelines to write this report.

We thank **Prof. Vishal S. Badgujar** BE project co-ordinator, Department of IT, APSIT for being encouraging throughout the course and for guidance.

We also thank the entire staff of APSIT for their invaluable help rendered during the course of this work. We wish to express our deep gratitude towards all our colleagues of APSIT for their encouragement.

Student Name1:Siddhesh Gaikwad
Student ID1:18104069

Student Name2:Krishita Tolia
Student ID2:18104021

Student Name3:Akshata Gawas
Student ID3:18104039

Declaration

We declare that this written submission represents our ideas in our own words and where others' ideas or words have been included, We have adequately cited and referenced the original sources. We also declare that We have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in our submission. We understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

(Signature)

(Siddhesh Gaikwad and 18104069)
(Krishita Tolia and 18104021)
(Akshata Gawas and 18104039)

Date:

Abstract

The food industry is expanding every day, and it is crucial to maintain the required standards which impact their market value. To maintain these standards, we use human resources, which are inconsistent, expensive, and time-consuming. With the help of automation of classification, we can speed up this process with less expensive resources using Computer Vision and the Internet of Things(IoT). The evolution of the Internet of Things(IoT) has played an essential role in making the devices more innovative and more connected. The large amount of data collected from these devices can be used for data analysis which helps the industries to plan their future decisions. This paper proposes the idea of implementing an infrastructure having a micro-controller that would accurately segregate three kinds of fruits into two categories, i.e., Fresh and Rotten. The classification will be done with the help of the Deep Learning algorithm, Convolutional Neural Network(CNN), by using a dataset containing images of those three fruits and considering the input from the sensors, which include sensors such as alcohol and methane sensors. The infrastructure proposed in the report considers the standards of Industry 4.0, which implies the real-world implementation of the infrastructure.

Contents

List of Figures

List of Abbreviations

IoT:	Internet of Things
CNN:	Convolution Neural Network
VGG:	Visual Geometry Group
RGB:	Red Green Blue

Chapter 1

Introduction

The fruit industry has become the third major after grain and vegetables. The speedy development of the fruit industry has brought us visible economic benefits, however conjointly brought us a series of drawbacks, fruit classification is one amongst them.

1.1 Preface

Retail and supermarkets need manual labor to sort and classify fruits depending on their freshness. This includes not simply labor costs but also the time spent on these operations. Thus, to overcome this problem, we need an automatic system that will scale down the efforts of humans, reduce the price of production and time of production. Several kinds of research are carried out that depicts the use of a convolutional neural network(CNN) on various aspects [2][3][4][14][19] and plenty of more. With the assistance of previous studies, we have proposed a model that can determine whether the fruit is fresh or rotten using a convolutional neural network(CNN). With the appliance of deep learning architecture within the field of image recognition, a convolutional neural network(CNN), known as one of the typical deep learning models used for transfer learning, encompasses a good performance in image classification. CNN has various architectures that are developed with the evolving technologies, one of which is Inception-V3. It is a convolutional neural network architecture from the inception family that makes many improvements, considering accuracy as well as computational cost [9] [10] [11]. Thus, Inception-V3 is appropriate for the fresh and rotten fruits dataset. Classification will be based on color, shape, edge detection, ethanol emission, methane emission, and texture. The Fourth Industrial Revolution, known as Industry 4.0, is the digital transformation of manufacturing and related industries and value creation processes. It involves the automation of traditional manufacturing and industrial practices using modern innovative technology. Considering the standards of industry 4.0 [5] [6], this project contains a conveyor belt together with the sensors such as methane, ethanol to detect spoilage of the fruit. Additionally, data analysis can offer information regarding the fruits that will successively state their impact on the production.

1.1.1 Key Objectives

- With the help of deep learning classify fruit into fresh and rotten.
- To achieve industry 4.0 standards by integrating machine learning and sensors.

Chapter 2

Literature Review

Year	Author	Problem Description	Dataset used	Algorithm	Learning Type	Accuracy	Limitations
2020	Sai Sudha Sonali Palakodati, Venkata RamiReddy Chirra	Fresh and Rotten Fruits Classification Using CNN and Transfer Learning	3 fruits: apples, bananas and oranges	CNN	Transfer Learning	97.82%	Small dataset with small number of convolution layers.
2015	Karen Simonyan, Andrew Zisserman	Very Deep Convolutional Networks for Large-Scale Image Recognition	ILSVRC-2012 dataset	CNN-VGG	Supervised Learning	top-5 test error: 6.8%	A fixed kernel size of 3x3 was used.
2020	Deepika Srinivasan, Mahmoud Yousef	Apple Fruit Detection and Maturity Status Classification	Kaggle's Fresh and Rotten fruits Image dataset	CNN-ResNet50	Supervised Learning	97.92%	Locally based system with small dataset.
2019	Mengying Shu	Deep learning for image classification on very small datasets using transfer learning	6000 images of dogs and cats.	CNN: VGGNet, GoogleNet, InceptionResNet	Transfer Learning	InceptionResNet: 96%, Inception V3: 95%	Problem of underfitting was observed.
2020	Yuhang Fu	Fruit Freshness Grading Using Deep Learning	6 fruits: apple, dragon fruit, kiwi, pear, banana, orange.	YOLO for detection and classification, CNN for freshness level regression	Supervised and Transfer Learning	91.49%	YOLOv3 localize objects through rectangular bounding boxes that results in background noises.

Figure 2.1: Literature review

We have reviewed various relevant research papers discussing the study of different Neural Network classification algorithms used in the past few years with their obtained accuracy. Among the recent ones, the study of Apple Fruit Detection and Maturity Status Classification [8] obtained the highest accuracy of 97.92% by fine-tuning the ResNet50 architecture of the CNN algorithm. But it had a limitation as the system was locally based, and the model was trained on a small dataset.

In another paper on the topic of Fresh and Rotten Fruits Classification using CNN and transfer learning [1], various architectures of CNN were used to check the accuracy. A simple CNN model was deployed having a small number of convolution layers, and the model was trained on a small dataset, and an accuracy of 97.82% was obtained

A study on the Classification of images having dogs and cats [16] was reviewed, various CNN architectures like VGGNet, GoogLeNet., and InceptionResNet were used, and it was observed that InceptionResNet gave an accuracy of 96%. In comparison, Inception V3 gave 95% accuracy. As the dataset was minimal, a problem of overfitting was observed in the study.

A study on Mango Classification System based on Machine Vision and Artificial Intelligence [1] was studied. They used an ANN algorithm to sort mangoes and detect blemishes by considering features such as size, density, color, etc. got an accuracy of 90%.

A paper on the topic Deep Features Based Approach for Fruit Disease Detection and Classification [20] has described the use of the CNN algorithm on Quadtree segmented RGB images for detecting the fruit disease with an accuracy of 93%. However, a fixed-size kernel was used in the algorithm.

A paper on the topic of High-performance vegetable classification from images based on the AlexNet deep learning model [19] was studied, the CNN algorithm AlexNet was used on the dataset ImageNet from which broccoli, pumpkin, cauliflower, mushrooms, and cucumber images were used, and it gave an accuracy of 92.1%.

However, as a small number of images were used, it may result in overfitting. So, we reviewed various algorithms and tried to overcome the limitations observed in the reviewed studies. Our dataset has an adequate number of images, and we wish to expand it as we deploy the architecture. The algorithm selected is Inception V3 which is considered one of the efficient CNN algorithms. It has varied kernel size, batch normalization, and Auxiliary branches, which will help conquer the problem of vanishing gradient. The software part of our proposed system will be deployed on the cloud, and the data obtained from the system will be used for data analysis.

2.1 Objectives

- Classification of fruits into fresh and rotten.
- To achieve high accuracy by integrating machine learning and sensors.
- To perform data analysis on classification of fruits.

.

Chapter 3

Project Design

3.1 System Architecture

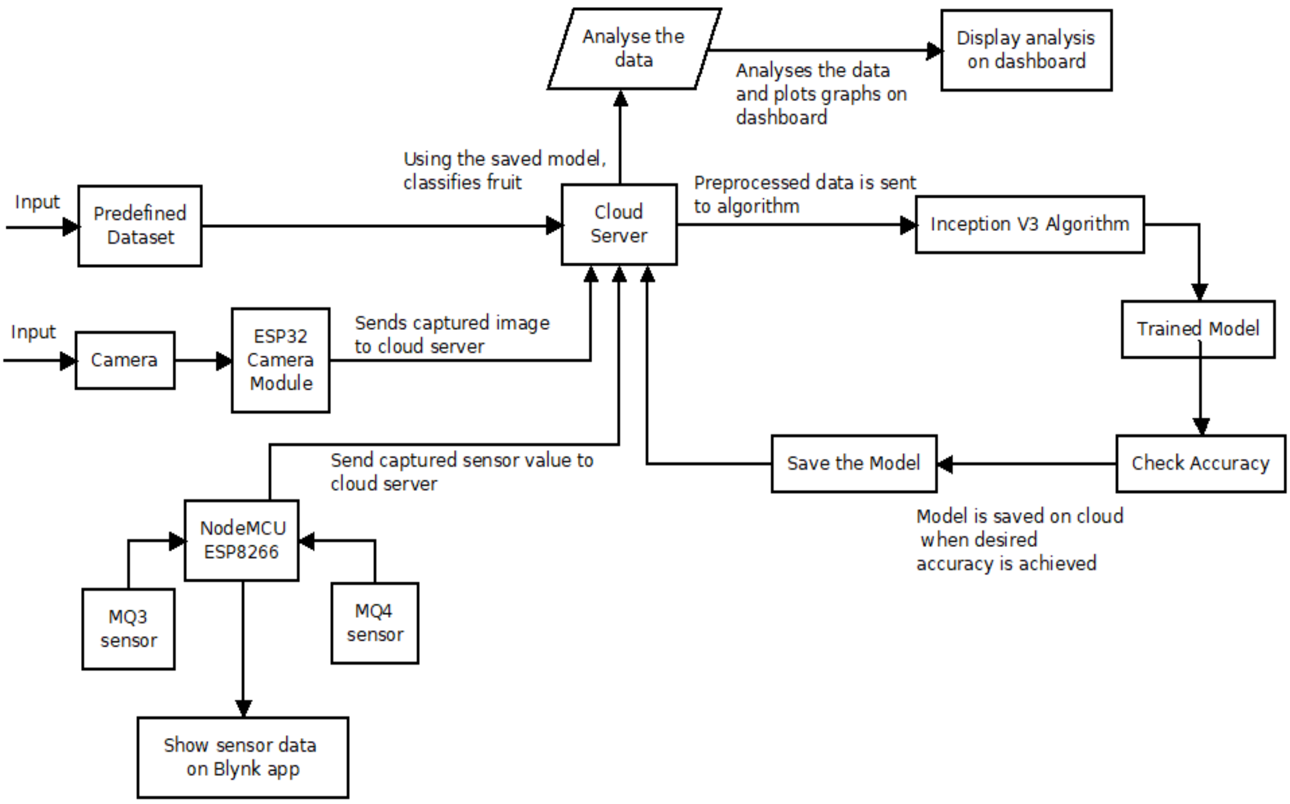

Figure 3.1: Demonstration of proposed architecture

The architecture consists of software as well as hardware. The spotlight of our project is to classify and segregate fruits according to their freshness. The paper proposes the implementation of Inception-V3, an advanced architecture of Convolutional Neural Network (CNN) using transfer learning [14] [16] [21] for image recognition and softmax function is used for image classification. For classification, we need a supervised learning model which will brief us on whether the fruit is fresh or rotten so, Fig no. 2 shows the whole process to be accomplished. Firstly, a predefined dataset stored in the database will be passed for preprocessing, including resizing, noise reduction. It is a crucial step to be performed before passing it for training. After that, the images will be passed to the model for training, and after accomplishing it, the trained model will be saved in the database having the desired accuracy. Libraries such as Keras, TensorFlow, NumPy are used for training the model.

For calculating accuracy, we pass the test set for prediction and check whether the predicted values match the original value. For training the model, dataset used is downloaded from Kaggle, and it consists of images of fresh apples, bananas, and oranges, and rotten apples, bananas, and oranges, i.e., it consists of 6 classes. 13,599 are the total number of images in the dataset, where 10901 images are used for training and 2698 for testing purpose. Along with this we have also created our own dataset of 634 images. Some of them are displayed in Fig. 3. First, data augmentation was performed by rescaling the image, applying shear, applying horizontal flip, and zooming into the image. Then the entire dataset is shaped to 224x224x3 and into a NumPy array for a more straightforward convolution process.

3.2 UML Diagram

3.2.1 Use Case Diagram

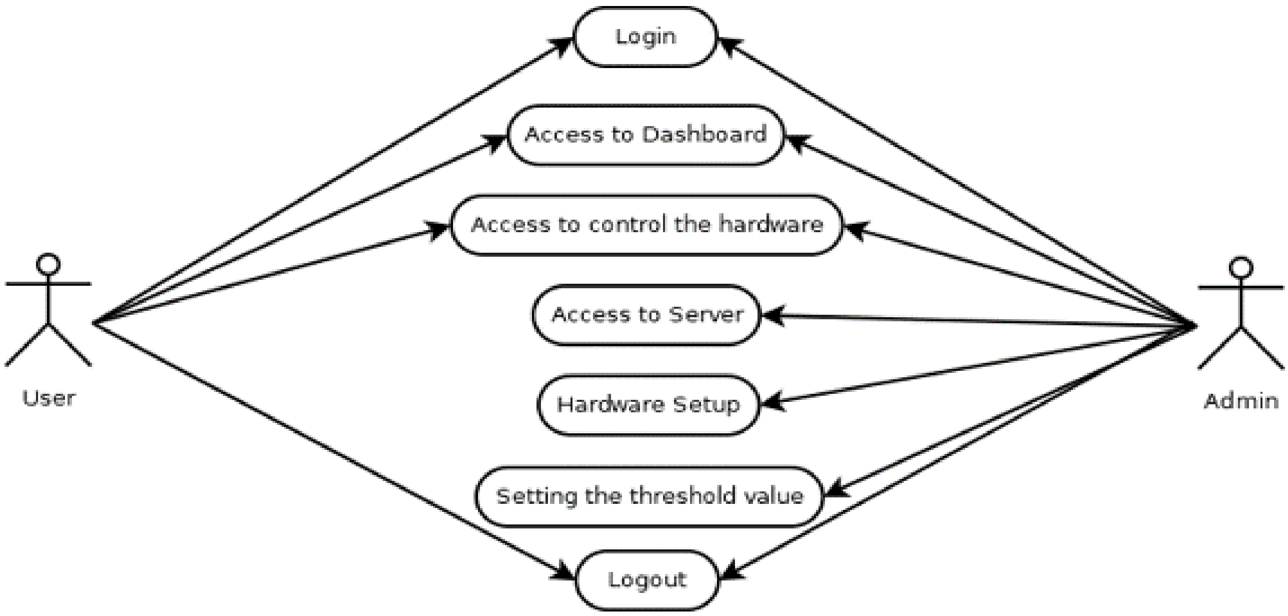

Figure 3.2: Use Case Diagram

3.2.2 Activity Diagram

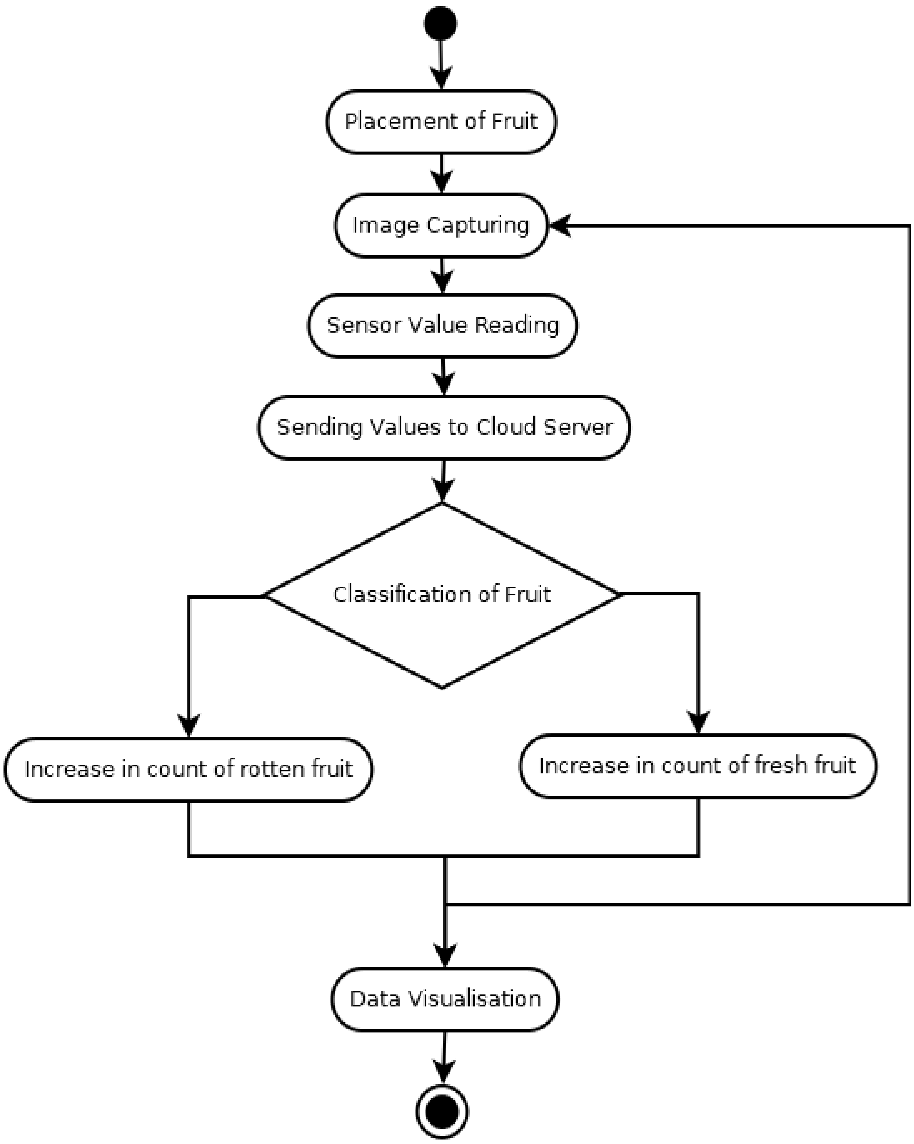

Figure 3.3: Activity Diagram

3.2.3 Sequence Diagram

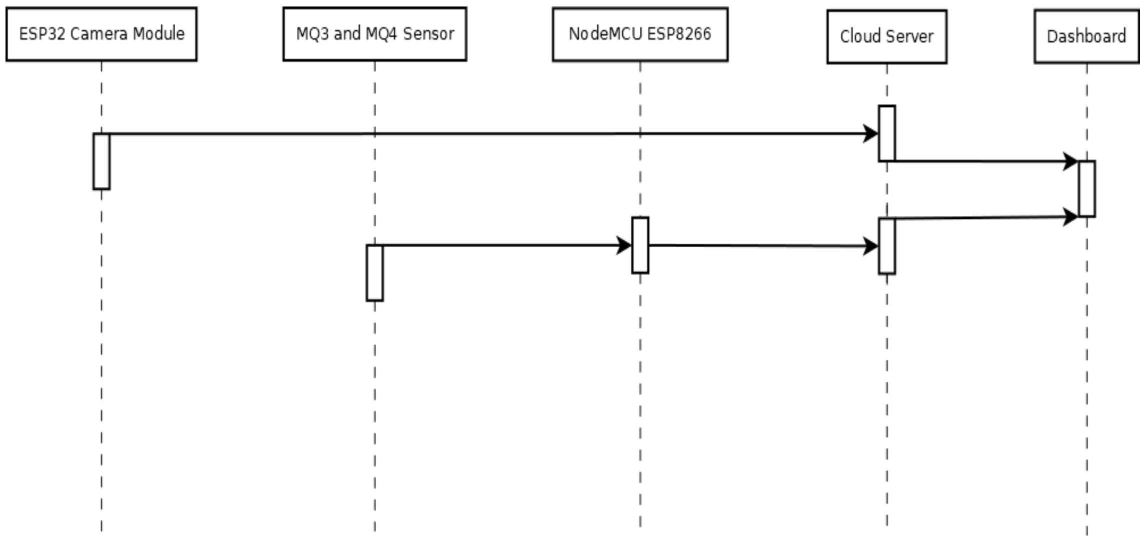

Figure 3.4: Sequence Diagram

3.3 Dataset

Figure 3.5: Creation of Dataset

The image shown above was taken when we were creating our own dataset. For creating our dataset,we used the same camera used in our proposed architecture. We captured 600 images of fruits including Apple, Banana and Orange.

Figure 3.6: Image of apple from dataset

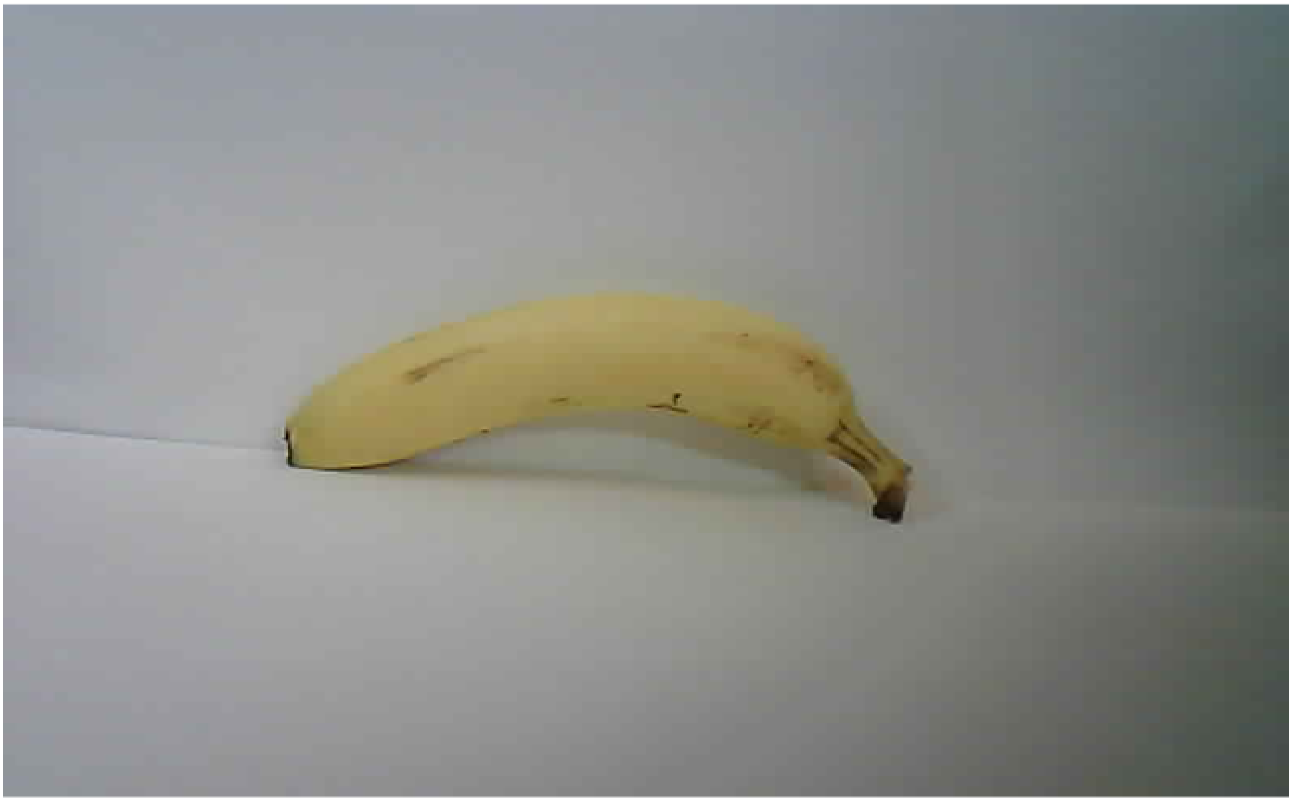

Figure 3.7: Image of banana from dataset

Figure 3.8: Image of orange from dataset

Chapter 4

Project Implementation

4.1 Hardware

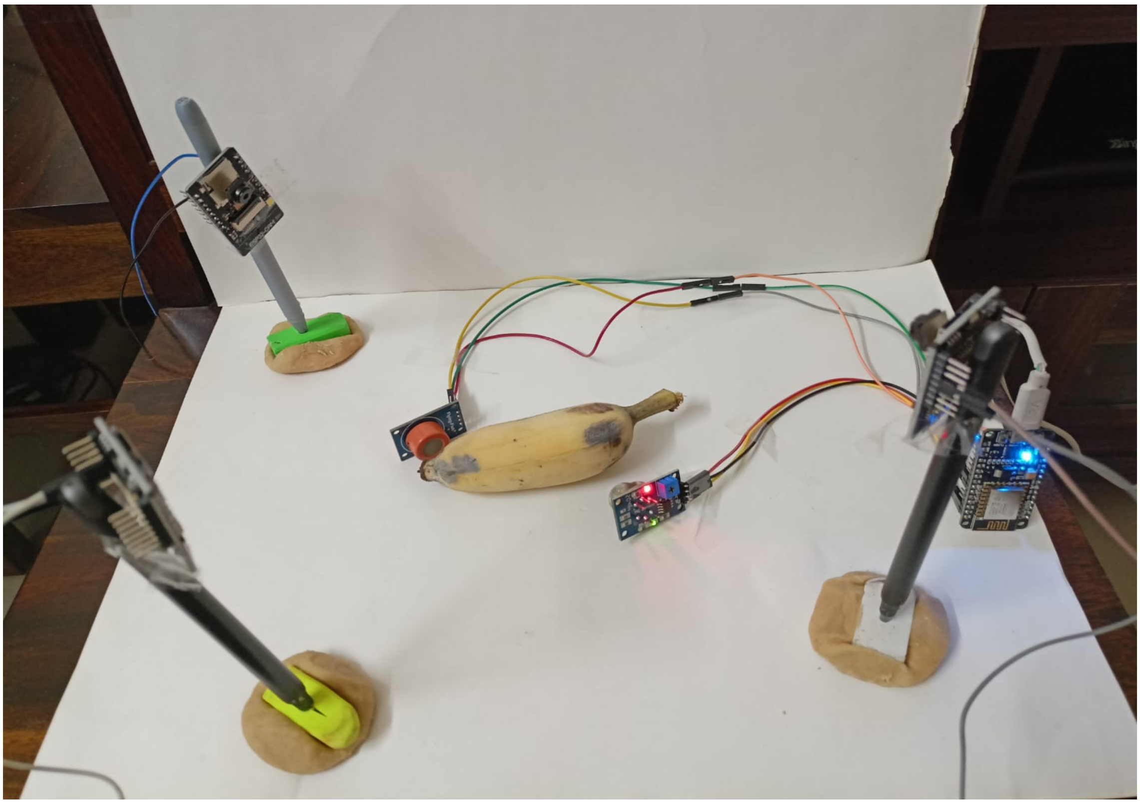

Figure 4.1: Hardware Setup

As mentioned in the project architecture, our hardware includes 3 ESP32 camera modules, MQ3 sensor and MQ4 sensor. The camera modules are programmed and then are placed in such position that it captures pictures of fruits from all directions. The live images are captured and are passed for classification.

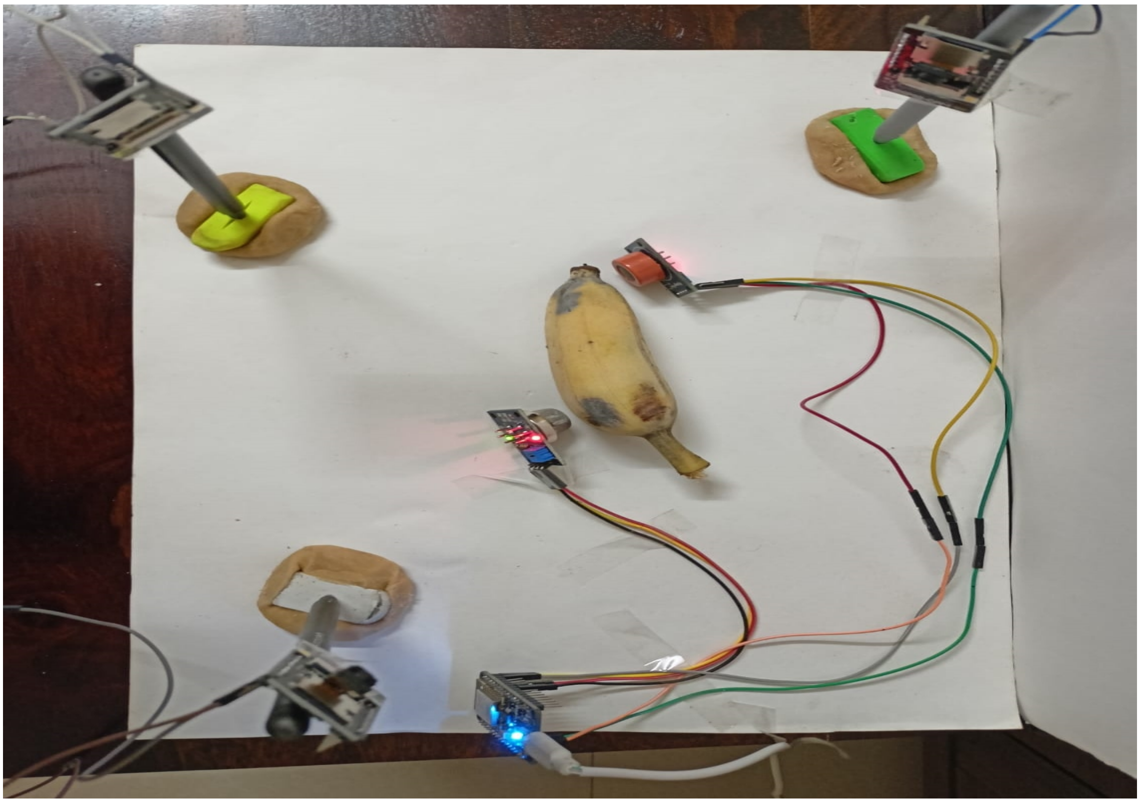

Figure 4.2: Hardware Setup

The sensors are placed closely to the fruit in order to collect data accurately and this data is collected using NodeMCU ESP8266.

4.2 Code Snippets

```
[3] import os
    import matplotlib.pyplot as plt
    import numpy as np
    import tensorflow as tf
    from tensorflow.keras.optimizers import RMSprop
    from tensorflow.keras import layers
    from tensorflow.keras import Model
    from tensorflow.keras.preprocessing.image import ImageDataGenerator
    from tensorflow.keras.utils import plot_model
    from tensorflow.keras import backend as K
    import matplotlib.pyplot as plt
    import shutil
    from tensorflow.keras.preprocessing import  image
    from tensorflow.keras.applications.inception_v3 import preprocess_input
```

Figure 4.3: Libraries imported

The code snippet shown below has all the libraries used in our code for various functionalities.

```
from tensorflow.keras.optimizers import RMSprop
from tensorflow.keras.applications import InceptionV3
from tensorflow.keras import layers
from tensorflow.keras import Model

pre_trained_model = InceptionV3(input_shape=(150,150,3),
                                include_top=False)

for layer in pre_trained_model.layers:
    layer.trainable = False

x = layers.Flatten()(pre_trained_model.output)
x = layers.Dense(1024, activation='relu')(x)
x = layers.Dropout(0.2)(x)
x = layers.Dense(6, activation='softmax')(x)
```

Figure 4.4: Importing inceptionV3 model

In the above image, we can see that we imported the pretrained InceptionV3 model having input shape 150,150,3. As we set the layer training to false, we freeze the model which in turn reduces the computational timing. Then we Flatten the last layer, add an activation function "relu" at first. Then a dropout layer with rate being 0.2 is added with a final dense layer having 6 output channels and activation function used is softmax.

```
model = Model(pre_trained_model.input, x)

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

Figure 4.5: Model Compilation

Here, we load the model by taking its input from the previously mentioned pre-trained model and the additional layers which were mentioned in the previous image. We compile the model using "Adam" optimizer, "categorical crossentropy" as the loss function and metric function used is "accuracy" which is used to judge the performance of our model.

```
callbacks = myCallback()
history = model.fit(
    train_generator,
    steps_per_epoch=(train_len/32),
    epochs=3,
    verbose=1,
    validation_data=validation_generator,
    validation_steps=(val_len/32),
    callbacks=[callbacks],
)
```

Figure 4.6: Model Training

In the above code snippet, we can see that we used the fit() function to train the model by giving 'train generator' as iterator for training set, 'validation generator' as iterator for testing or validation set, 3 epochs, validation steps as length of validation step divided by 32 and used a callback function. This callback function runs the training of model until a desired accuracy is achieved.

4.3 Dashboard

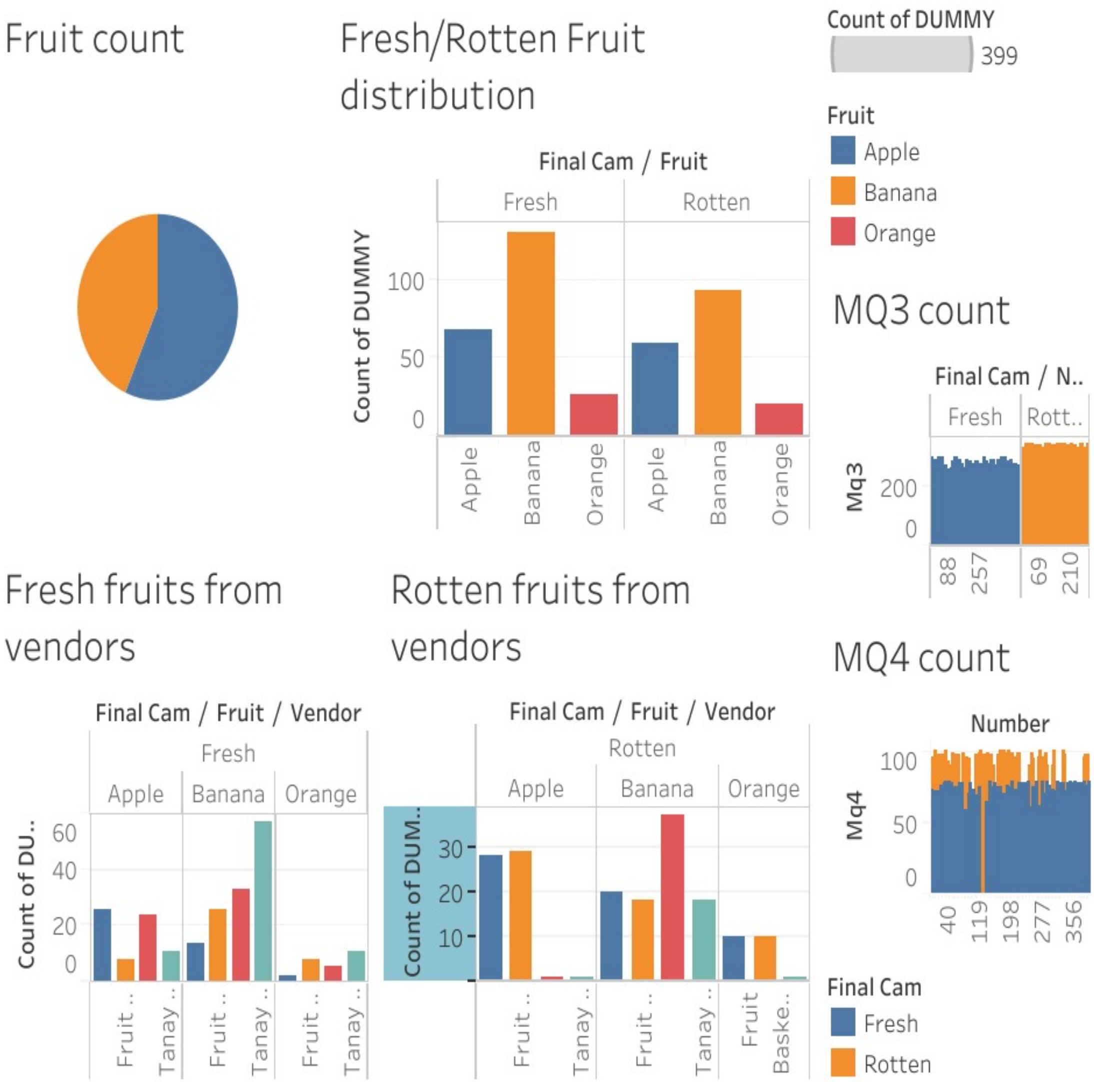

Figure 4.7: Importing inceptionV3 model

The above figure shows an interactive dashboard where we can see the statistics of the segregation of fruits in various sections. The first sections shows the count of Fruits divided into fresh and rotten. The second section shows the fruit specific count of fresh and rotten fruits. There is also a section where the fruits from different vendors are analysed. MQ3 and MQ4 range for fresh and rotten fruits is also plotted in a graph.

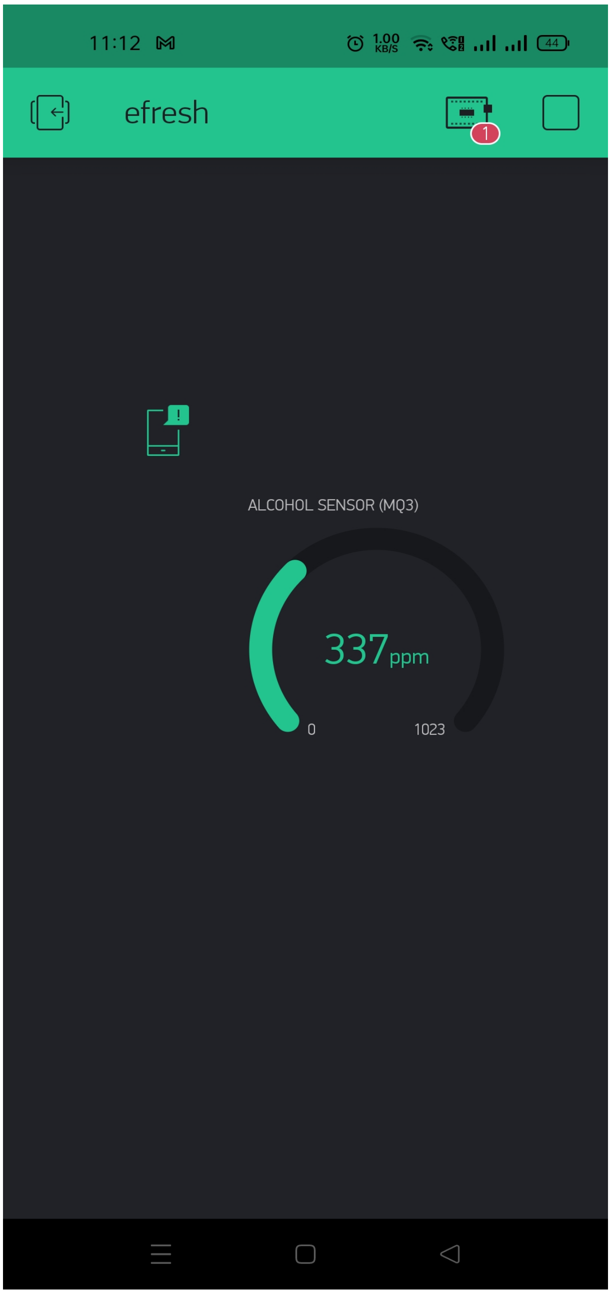

Figure 4.8: Mq3 Sensor Reading

This image shows the real-time MQ3 sensor reading which we obtained from ESP8266 and is displayed on the Blynk app.

Chapter 5

Testing

5.1 Unit Testing

UNIT TESTING is a type of software testing where individual units or components of a software are tested. The purpose is to validate that each unit of the software code performs as expected. It is a very useful technique that can help you prevent obvious errors and bugs in your code. It involves testing individual units of the source code, such as functions, methods, and class to ascertain that they meet the requirements and have expected behaviour. Unit tests are usually small and don't take much time to execute. In our case we began writing code for our machine learning algorithm and sensors in form of units. We then tested each unit separately to minimize the error. Isolating the code helps in revealing unnecessary dependencies between the code being tested and other units or data spaces in the product. These dependencies can then be eliminated.

5.2 Integration Testing

Integration testing is defined as a type of testing where software modules are integrated logically and tested as a group. A typical software project consists of multiple software modules, coded by different programmers. The purpose of this level of testing is to expose defects in the interaction between these software modules when they are integrated. Similarly we divided our code into single unit. After each unit has been thoroughly tested, it is combined with other units to form modules. This testing facilitates smooth integration of software and hardware, if any error is encountered it will be resolved for that specific unit.

Chapter 6

Result

Our main aim is to create a automated segregation tool for fruit industry which can be used in place of human labor. The tool will also provide important insight using the data captured during segregation.

```
loss, acc = model.evaluate(test_generator,steps = (nb_samples), verbose=1)
print('accuracy test: ',acc)

Found 2705 images belonging to 6 classes.
2705/2705 [==============================] - 198s 73ms/step - loss: 0.1347 - accuracy: 0.9523
accuracy test:  0.9523105621337891
```

Figure 6.1: Accuracy

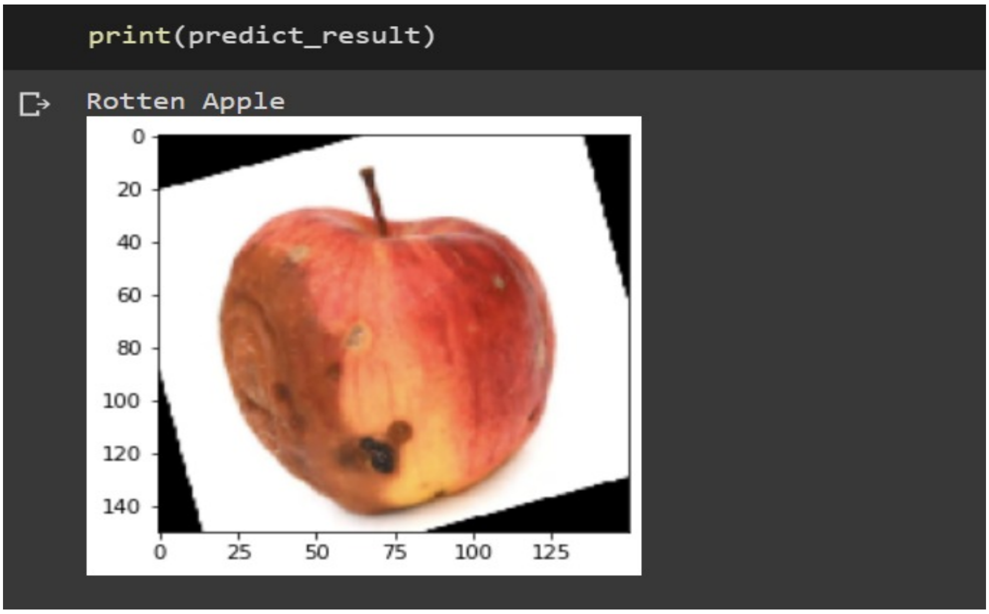

Figure 6.2: Predicted Value

Chapter 7

Conclusions and Future Scope

7.1 Conclusion

The process of recognizing, classifying, and segregating fruits according to their freshness is proposed in this paper. CNN architecture, i.e., Inception-V3 using transfer learning, is implemented in this proposed framework. Datasets with different fruits such as apple, banana, orange are used to train and test the model. The accuracy of the trained model with epochs 100 and batch-size 16 was recorded as 99.17. A touch of the Internet of things (IoT) is given to the architecture by using sensors, camera modules and sending data over the internet to the trained model stored on a cloud. Sensors give an additional advantage along with image classification to filter fresh and rotten fruits. While designing the hardware model, the standards of Industry 4.0 were kept in consideration. This proposed framework would aid the food industries as a considerable amount of capital and time is spent on labor-intensive and repetitive tasks.So to conquer the drawbacks, this proposed framework can be implemented. Data analysis will provide information about the fruits, which will, in turn, state their impact on the production. Further, the analytical report will be prepared on the web portal.

7.2 Future Scope

- By giving large number of relevant data will increase the accuracy of the system.
- Camera with advance specifications will increase the accuracy as compared to the current system.
- We can further include more varieties of fruits for classification.
- Conveyor belt could be used to automate the process.

Bibliography

[1] Nguyen Truong Thinh, Nguyen Duc Thong, and Huynh Thanh Cong, “Mango Classification System Based on Machine Vision and Artificial Intelligence,” Ho Chi Minh City University of Technology, VNU HCMC, 2019 IEEE 7th International Conference on Control.

[2] Rajesh Yamparala, Ramaiah Challa, V.Kantharao, P.Seetha Rama Krishna, ”Computerized Classification of Fruits using Convolution Neural Network”, IEEE 7th International Conference on Smart Structures and Systems ICSSS 2020.

[3] M. Shamim Hossain , Senior Member, IEEE, Muneer Al-Hammadi and Ghulam Muhammad , Member, IEEE, “Automatic Fruit Classification Using Deep Learning for Industrial Applications,” IEEE transactions on industrial informatics, vol. 15, no. 2, February 2019.

[4] Liuchen Wu, Hui Zhang, Ruibo Chen, Junfei Yi, ”Fruit Classification using Convolutional Neural Network via Adjust Parameter and Data Enhancement”, 12th International Conference on Advanced Computational Intelligence (ICACI).

[5] Ms. Jayshree C. Phad, Ms.Kajal A. Nagare, Ms. Shubhangi R. Kalokhe, Prof. A. R. Khandare, Prof. D. Y. Thorat5, ”Conveyor Belt Fault Detection Using IOT”, 2019 IJRTI — Volume 4, Issue 3 — ISSN: 2456-3315.

[6] G. Lodewijks, W. Li, Y. Pang, X. Jiang, ”An Application of the IoT in Belt Conveyor Systems”.

[7] Anuja Bhargava, Atul Bansal ”Fruits and vegetables quality evaluation using computer vision: A review”, Journal of King Saud University – Computer and Information Sciences(2018).

[8] Deepika Srinivasan, Mahmoud Yousef, ”Apple Fruit Detection and Maturity Status Classification”, International Journal of Recent Technology and Engineering (IJRTE) ISSN: 2277-3878, Volume-9 Issue-2, July 2020.

[9] CHENG WANG, DELEI CHEN, LIN HAO, XUEBO LIU, YU ZENG, JIANWEI CHEN, AND GUOKAI ZHANG, ”Pulmonary Image Classification Based on Inception-v3 Transfer Learning Model”, Digital Object Identifier 10.1109/ACCESS.2019.2946000.

[10] Vincent Vanhoucke, Sergey Ioffe, Jonathon Shlens, ”Rethinking the Inception Architecture for Computer Vision”, arXiv:1512.00567v3 [cs.CV] 11 Dec 2015.

[11] Qing Guan, Xiaochun Wan, Hongtao Lu, Bo Ping, Duanshu Li1, Li Wang, Yongxue Zhu, Yunjun Wang1, Jun Xia,”Deep convolutional neural network Inception-v3 model for

differential diagnosing of lymph node in cytological images: a pilot study", Ann Transl Med 2019;7(14):307 — http://dx.doi.org/10.21037/atm.2019.06.29

[12] Sergey Ioffe, Christian Szegedy, "Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift", arXiv:1502.03167 11 Feb 2015.

[13] Christian Szegedy, Wei Liu, "Going deeper with convolutions", arXiv:1409.4842v1 [cs.CV] 17 Sep 2014

[14] Sai Sudha Sonali Palakodati, Venkata RamiReddy Chirra,"Fresh and Rotten Fruits ClassificationUsing CNN and Transfer Learning"

[15] Karen Simonyam, Andrew Zisserman,"Very Deep Convolutional Networks for Large-Scale Image Recognition"

[16] Mengying Shu,"Deep learning for image classificationon very small datasets using transfer learning"

[17] Yuhang Fu,"Fruit Freshness Grading Using Deep Learning"

[18] Jagadeesh Devdas Pujari, Rajesh Yakkundimath, "Grading and Classification of Anthracnose Fungal Disease of Fruits based on Statistical Texture Features"

[19] Ling ZhuZhenbo Li, Chen Li, Jing Wu, Jun Yue, "High performance vegetable classification from images based on AlexNet deep learning model"

[20] Ranjit K N, Raghunandan K S, Naveen C, Chethan H K, Sunil C, "Deep Features Based Approach for Fruit Disease Detection and Classification"

[21] Ribani, Ricardo and Marengoni, Mauricio, "A survey of transfer learning for convolutional neural networks",2019 32nd SIBGRAPI Conference on Graphics, Patterns and Images Tutorials (SIBGRAPI-T)

[22] Channe, Hemlata and Kothari, Sukhesh and Kadam, Dipali, "Multidisciplinary model for smart agriculture using internet-of-things (IoT), sensors, cloud-computing, mobile-computing & big-data analysis", Int. J. Computer Technology & Applications

[23] Naveed Shahzad, Usman Khalid, Atif Iqbal, Meezan-Ur-Rahman, "eFresh – a Device to Detect Food Freshness", International Journal of Soft Computing and Engineering (IJSCE)

Appendices

Detailed information, lengthy derivations, raw experimental observations etc. are to be presented in the separate appendices, which shall be numbered in Roman Capitals (e.g. "Appendix I"). Since reference can be drawn to published/unpublished literature in the appendices these should precede the "Literature Cited" section.

Appendix-A: Installing libraries

1. TensorFlow is a free and open-source software library for machine learning and artificial intelligence. It can be used across a range of tasks but has a particular focus on training and inference of deep neural networks.
!pip install tensorflow

2. Keras is an open-source software library that provides a Python interface for artificial neural networks. Keras acts as an interface for the TensorFlow library.
!pip install keras

3. grpcio is the gRPC package for Python.gRPC is a modern open source high performance Remote Procedure Call (RPC) framework that can run in any environment.
!pip install grpcio

4. Python pickle module is used for serializing and de-serializing a Python object structure. Any object in Python can be pickled so that it can be saved on disk.
!pip install pickle

5.Pandas is an open source Python package that is most widely used for data science/data analysis and machine learning tasks.
!pip install pandas

6. NumPy, which stands for Numerical Python, is a library consisting of multidimensional array objects and a collection of routines for processing those arrays.
!pip install numpy

Publication

Paper entitled **"E-fresh : Computer vision and IOT based system for food industry"** is presented at **"International Conference on Advances in Computing, Communication and Control (ICAC3 2021)"** by **"Krishita Tolia, Akshata Gawas, Siddhesh Gaikwad".**