Environmental Science Project Report

on

AI based Rotten Fruit Detection and Segregation

Submitted by

Shivam Deolankar(2023IMT-073)
Sumit Chintanwar(2023IMT-082)
Taksh Patel(2023IMT-083)
Udit Shrivastava(2023IMT-084)
Yash Wani(2023IMT-087)

Submitted to

Prof. Anurag Srivastava

विश्वजीवनामृतं ज्ञानम्

ABV-Indian Institute of Information Technology Management
Gwalior

Date: 16/04/2024

Contents

Abstract

Detection and classification of rotten fruits is an important problem in the agriculture and food industry and has a serious impact. Food quality, safety and waste reduction. The project aims to create an automated system that uses advanced technologies such as image processing, machine learning and sensor integration to instantly and accurately identify and classify rotten fruit.

The proposed system uses state-of-the-art computer vision algorithms and deep learning models to analyze data obtained from special cameras and sensors. By extracting and processing important features, including color, texture, and morphological characteristics, the system can reliably distinguish between fresh fruit and fresh fruit. Additionally, the integration of machine learning enables continuous improvement of the system's performance by providing diverse data and adaptive learning. High precision coating in decay. The system's performance is calculated using industry standard metrics such as precision, recall, and F1 score, ensuring its reliability and usability in real-world situations. It aims to improve efficiency and good management practices in the fruit industry to reduce food waste, increase food safety and improve consumption resources. Additionally, design and construction methods can be easily integrated into existing agricultural and food projects, promoting widespread and relevant development. The program contributes to the widespread use of sustainable and innovative technologies in agriculture and food, in line with international initiatives to improve the food safety of rice, promote resource conservation and reduce the environmental impact of food waste.

1 Introduction

1.1 Background

The product focuses on addressing the critical issue of detecting and segregating rotten fruits in the agricultural and food processing industries. Rotten fruits not only compromise food quality and safety but also contribute significantly to food waste. To combat this problem, our Product aims to develop an automated system utilizing advanced technologies such as image processing, machine learning, and sensor integration.

By accurately identifying and segregating rotten fruits in real-time, the system promises to enhance efficiency, reduce waste, and improve overall quality management in the fruit supply chain. This introduction sets the stage for a comprehensive exploration of our Product's objectives, methodologies, and anticipated outcomes.

Furthermore, Our Product aligns with broader efforts towards sustainability and technological innovation in agriculture and food production. By leveraging cutting-edge technologies and interdisciplinary approaches, we seek to contribute to the global agenda of reducing food waste, enhancing food safety, and optimizing resource utilization. Through collaboration and innovation, our Product aims to make a tangible impact on the efficiency and sustainability of fruit quality management practices, benefiting both industry stakeholders and consumers alike

1.2 Objectives

1. To develop an automated system capable of accurately detecting and segregating rotten fruits from fresh ones, leveraging advanced computer vision and machine learning techniques. The system will employ image processing algorithms and deep learning models to analyze visual data captured by specialized cameras and sensors, extracting relevant features such as color, texture, and morphological characteristics.

2. To achieve high accuracy in rotten fruit detection and segregation across various fruit varieties, contributing to reduced food waste, improved food safety, and optimized resource utilization within the agricultural and food processing industries. Extensive experimentation and evaluation will be conducted to quantify the system's performance using industry-standard metrics.

3. To design a modular and scalable architecture integrating seamlessly into existing agricultural and food processing workflows for widespread adoption. Aligning with sustainability efforts, it promotes resource conservation and mitigates environmental impact of food waste.

2 Prototype Details

2.1 Components

1. **Arduino Uno/Arduino Pro Micro**
The Arduino Uno and Arduino Pro Micro are popular microcontroller boards widely used in electronics projects and prototyping. The Arduino Uno features the ATmega328P microcontroller and offers a variety of digital and analog input/output pins, making it versatile for a wide range of projects. It is beginner-friendly, with a simple design and easy-to-use interface. The Arduino Pro Micro, on the other hand, is a compact board based on the ATmega32U4 microcontroller. It offers built-in USB connectivity and a smaller form factor, making it suitable for projects where space is limited. Both boards support the Arduino programming language and IDE, and are commonly used in robotics, home automation, IoT devices, and wearable technology.

2. **Servo Motors**
Servo motors are rotary actuators that allow precise control of angular

position. They consist of a motor, a control circuit, and a feedback mechanism. Key features include:

(a) Precision Control: Servo motors provide accurate and precise control over the rotation angle.

(b) Feedback Mechanism: A feedback mechanism, such as a potentiometer or encoder, provides information about the motor's current position.

(c) High Torque: Servo motors can provide high torque output relative to their size.

(d) Speed Control: Servo motors can be controlled to rotate at different speeds, allowing for dynamic movement.

(e) Wide Range of Applications: Servo motors are used in robotics, RC vehicles, industrial automation, 3D printers, and camera stabilization systems.

3. **Jumper Wires, Cardboard, and Miscellaneous Materials**
   Jumper wires are essential components in electronics and prototyping, used to create connections between various components on a breadboard or circuit board. They consist of flexible wires with connectors at each end, allowing for quick and easy connections without soldering. Cardboard and various stationary materials like fevicol are required to create a prototype for the segregation model.

2.2 Methodology

2.2.1 Data Preparation

1. **Data Collection:** Gather a dataset of images containing both fresh and rotten fruits. This dataset should be diverse, containing various types of fruits and varying degrees of freshness/rottenness.

2. **Data Preprocessing:** Preprocess the images to standardize their size, color, and orientation. This step may involve resizing, cropping, and normalizing the images.

3. **Labeling:** Annotate the images to indicate whether each fruit is fresh or rotten. This step is crucial for supervised learning.

4. **Split Data:** Divide the dataset into training, validation, and testing sets. Typically, you'll use around 70-80% for training, 10-15% for validation, and 10-15% for testing.

2.2.2 Model Development

1. **Model Selection:** Choose a suitable model architecture for your task. Convolutional Neural Networks (CNNs) are commonly used for image clas-

sification tasks. You can use pre-trained models like VGG, ResNet, or Inception and fine-tune them for your specific task.

2. **Model Training:** Train your chosen model using the training data. During training, the model learns to differentiate between fresh and rotten fruits based on the features extracted from the images.

2.2.3 Implementation

- **Code:** Code will play an important role in the operation of hardware, as the hardware prototype will work by instructions given by code via Arduino.

- **Hardware:** The hardware will consist of a cardboard-based prototype for segregation, functioning with Arduino and servo motors.

2.2.4 Evaluation and Fine-tuning

4. **Validation:** Evaluate the model's performance using the validation set. This step helps you tune hyperparameters and prevent overfitting.

5. **Testing:** Assess the model's performance on the testing set to get an unbiased estimate of its accuracy.

6. **Fine-tuning:** Depending on the performance of your model, you might need to fine-tune it by adjusting parameters, trying different architectures, or augmenting the data.

2.2.5 Circuit Diagram

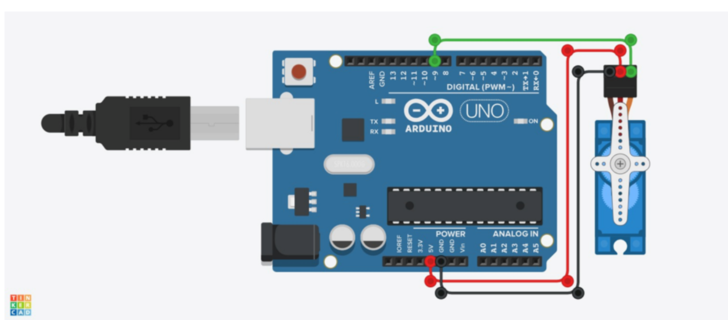


Figure 1: Arduino Uno board with servo motor connections

2.2.6 Tools and Libraries

In this topic, we are going to learn about the libraries used for this project.

No.	Library Used	Usage
1	Tensorflow	Used for deep learning tasks like creating model, prediction, etc.
2	Keras	Used for deep learning tasks like creating model, prediction, etc.
3	NumPy	Used for image matrix handling
4	OpenCV	Used for realtime detection of the fruit
5	Pandas	Data processing
6	Matplotlib	Used for rendering graphs for the model
7	PySerial	Used for communication with the Arduino

Figure 2: **Following figure shows the features of Model used for rotten fruit (Organic Matter) detection, it is AI based model.**

2.2.7 How detection works

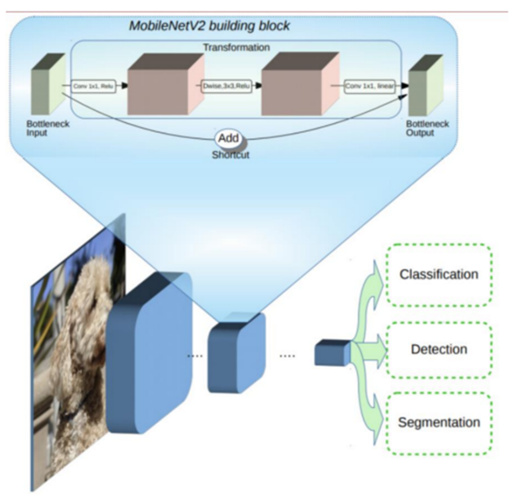
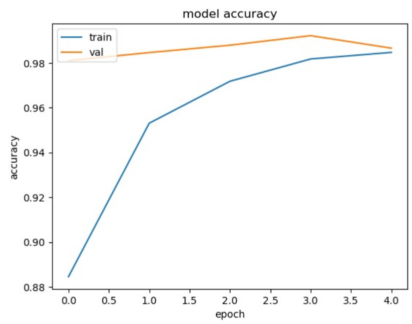


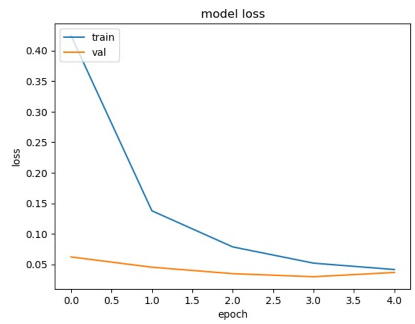
Figure 3: MobileNetV2 CNN architecture for rotten fruit detection and segmentation.

2.3 Model Accuracy and Loss

2.3.1 Fruit Detection Model

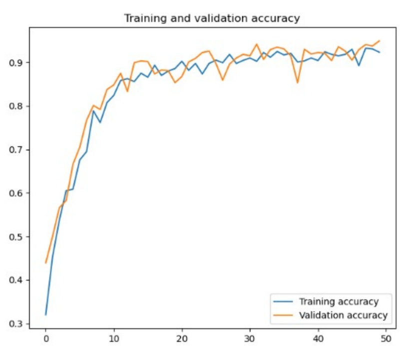


(a) Fruit Detection Model Accuracy



(b) Fruit Detection Model Loss

2.3.2 Rotten Fruit Detection Model



(a) Training and Validation Accuracy

(b) Training and Validation Loss

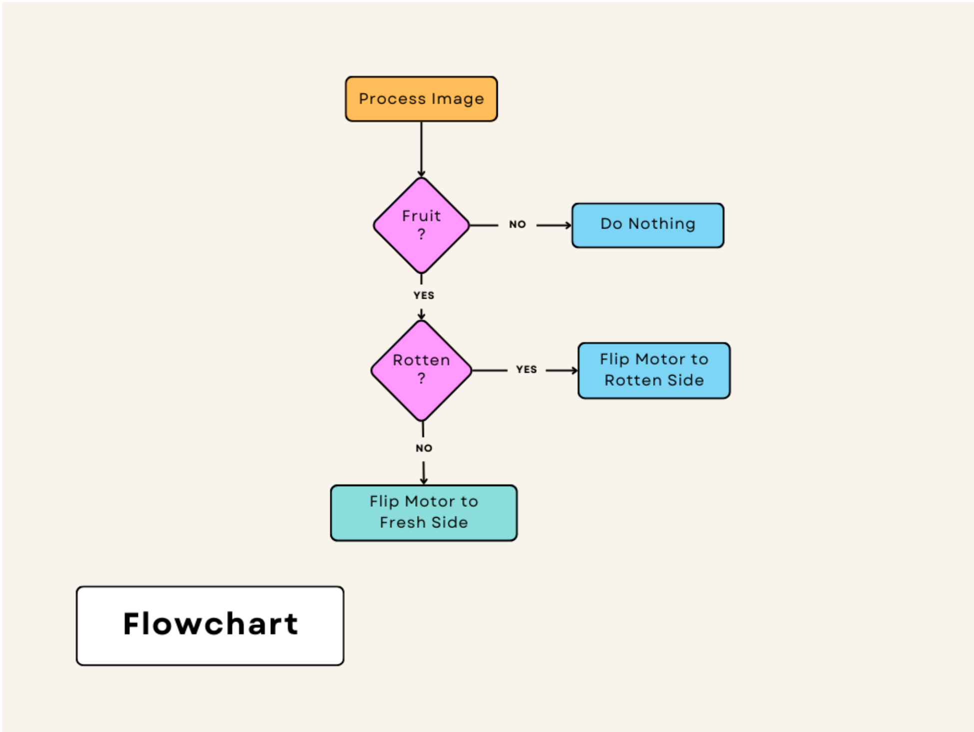


Figure 6: How our project works

2.4 Software

Our AI model for rotten fruit detection is hosted on GitHub at `https://github.com/udit2303/EES-Project`. This repository contains the source code, pre-trained models, and necessary files for deploying and utilizing our deep learning-based solution for accurately detecting and classifying rotten fruits.

The core of our model is built upon the MobileNetV2 convolutional neural network architecture, which has been fine-tuned and optimized for the specific task of rotten fruit detection and segmentation. The repository includes the Python scripts for training and evaluating the model, as well as the pre-trained weight files for immediate use.

Additionally, the repository provides a comprehensive README file with detailed instructions on setting up the environment, running the model, and interpreting the results. Sample input and output images are also included for testing and demonstration purposes.

The code is well-documented and follows best practices for maintainability and extensibility. We have also included a detailed explanation of the model architecture, training process, and evaluation metrics used for assessing the model's performance.

Feel free to explore the repository, clone or fork it, and contribute to further enhancing our rotten fruit detection solution.

3 Period vs Activity

Table 1: Project Timeline

Time Period	**Activity Completed**
15/03/2024	Project Acceptance
15/03/2024 - 17/03/2024	Research Phase
20/03/2024 - 05/04/2024	Development, Learning and Research Phase
07/04/2024 - 09/04/2024	Hardware Development and Training of AI model
10/04/2024	Hardware Integration
11/04/2024	Alpha Testing and Spotting of errors
12/04/2024	Rectifying Errors in Functioning of Segregator and Model
13/04/2024	Finishing Touches and Finalised Product

4 Challenges faced

4.1 During the development

1. **Technical Issues:** Technical problems such as compatibility issues between components, software bugs, or hardware malfunctions.

2. **Resource Constraints:** Limited availability of resources such as time, budget, or equipment.

3. **Scope Creep:** The project scope expanded beyond initial expectations, leading to increased complexity, time constraints, and resource allocation challenges.

4. **Lack of Expertise:** Insufficient expertise or experience in certain areas of the project resulted in difficulties in implementation or decision-making.

5 Contributions of Project Members

- **Shivam Deolankar:** Hardware Integration and Research
- **Sumit Chintanwar:** Software Development for Segregation
- **Taksh Patel:** Software Development related to training
- **Udit Shrivastava:** Software Development for Detection
- **Yash Wani:** Hardware Development for Segregation

6 Learning of Project Members

- **Shivam Deolankar:** Learning about available methods for development of AI Model and learning AIML basics.
- **Sumit Chintanwar:** Communication of arduino with other devices over USB.
- **Taksh Patel:** Data processing and choosing appropriate dataset and latex.
- **Udit Shrivastava:** Learning about different models and image processing.
- **Yash Wani:** Learning about various application of hardware components used in project and latex.

7 References

- `https://www.kaggle.com/code/anshuls235/pytorch-starter`
- `https://www.kaggle.com/code/jyotichoudhary5996/fresh-or-rotten-fruit-classifier`