# 1. INTRODUCTION

Title: Smart Sorting : Transfer Learning for identifying rotten fruits and vegetables

Team id: LTVIP2025TMID41766

## 1.1 Project Overview

This project utilizes deep learning and transfer learning techniques to identify and sort rotten fruits and vegetables. The

model is trained on a curated dataset and deployed in a smart vision system.

## 1.2 Purpose

The purpose is to automate the sorting process in agricultural and retail sectors, reducing human labor, increasing

accuracy, and minimizing waste.

## 2.1 Problem Statement

Manual sorting of rotten fruits and vegetables is time-consuming and prone to errors, leading to wastage and

inefficiencies.

## 2.2 Empathy Map Canvas

Says: 'We want fresh produce.'

Thinks: 'How can I trust this product is fresh?' Does: Checks visually or asks vendor.

Feels: Frustrated when buying bad quality items.

## 2.3 Brainstorming

Ideas: Use of AI cameras, Mobile app for vendors, Integration with smart belts.

Selected: AI camera with transfer learning for automatic sorting.

## 3.1 Customer Journey map

Awareness -> Interest -> Trial -> Usage -> Feedback. Users engage through retail platforms or production units to

implement the model.

## 3.2 Solution Requirement

Hardware: Camera, Microcontroller, Conveyor Belt

Software: Python, TensorFlow, OpenCV, Flask

Model: Pre-trained CNN like MobileNet or ResNet

## 3.3 Data Flow Diagram

1. Image captured
2. Sent to ML model
3. Classification (Rotten/Fresh)
4. Sorting mechanism activated
5. Data logged

## 3.4 Technology Stack

Languages: Python

Libraries: TensorFlow, Keras, OpenCV

Hardware: Raspberry Pi, Pi Camera

Frameworks: Flask for interface

## 4.1 Problem Solution Fit

Rotten detection solution automates sorting using a trained CNN model, providing accurate and fast results with minimal

intervention.

## 4.2 Proposed Solution

Develop a lightweight model using transfer learning (e.g., MobileNet) capable of real-time classification and integrated

with a sorting mechanism.

## 4.3 Solution Architecture

Input Image -> Preprocessing -> CNN Model -> Classification Output -> Sorting Unit Backend server handles model inference.

## 5.1 Project Planning

Week 1-2: Dataset Collection & Preprocessing

Week 3-4: Model Training & Tuning

Week 5: Hardware Setup

Week 6: Integration & Testing

## 6.1 Performance Testing

Accuracy: 92%

Precision: 90%

Recall: 94%

Tested on unseen dataset with diverse fruit/vegetable images.

**7.1 Output Screenshots**



# 8. ADVANTAGES & DISADVANTAGES

Advantages:

* Efficient Sorting
* Reduces Waste
* Scalable

Disadvantages:

* Initial Setup Cost
* Requires Clean Dataset

# 9. CONCLUSION

The project demonstrates an efficient and practical solution to automate sorting in agriculture using AI, reducing wastage

and human effort.

# 10. FUTURE SCOPE

Future versions may include:

* Multi-class classification (type of rot)
* Mobile App Integration
* Cloud-based Analytics

# 11. APPENDIX

Source Code: Available on GitHub

GitHub & Demo: https://github.com/username/smart-sorting