

Project Design Phase

Solution Architecture

Date	15 February 2025
Team ID	LTVIP2025TMID35093
Project Name	mart Sorting & Transfer Learning for Identifying Rotten Fruits and Vegetables
Maximum Marks	4 Marks

Solution Architecture:

Overview

This solution architecture bridges the gap between the **problem of manual fruit and vegetable sorting** and a **smart, AI-powered sorting system** using **transfer learning and embedded hardware**. The system automates the identification and separation of rotten items from fresh ones in real time.

Goals of the Solution Architecture

- Automate the detection of spoiled fruits/vegetables using AI.
- Reduce food wastage and improve quality control.
- Provide real-time results using edge devices.
- Minimize manual effort and human error.

Components of the Solution

1. Image Acquisition

- a. Camera Module (e.g., Pi Camera or USB Cam) captures images on a conveyor belt or sorting table.

2. Edge Device

- a. Raspberry Pi or Jetson Nano runs the trained transfer learning model.
- b. Handles real-time image processing and prediction.

3. AI Model

- a. MobileNetV2 (or ResNet50) with transfer learning classifies images as *fresh* or *rotten*.
- b. Model is trained using a labeled dataset of fruit and vegetable images.

4. Sorting Mechanism

- a. Servo/motor-based mechanism moves items based on classification result (e.g., fresh → left bin, rotten → right bin).

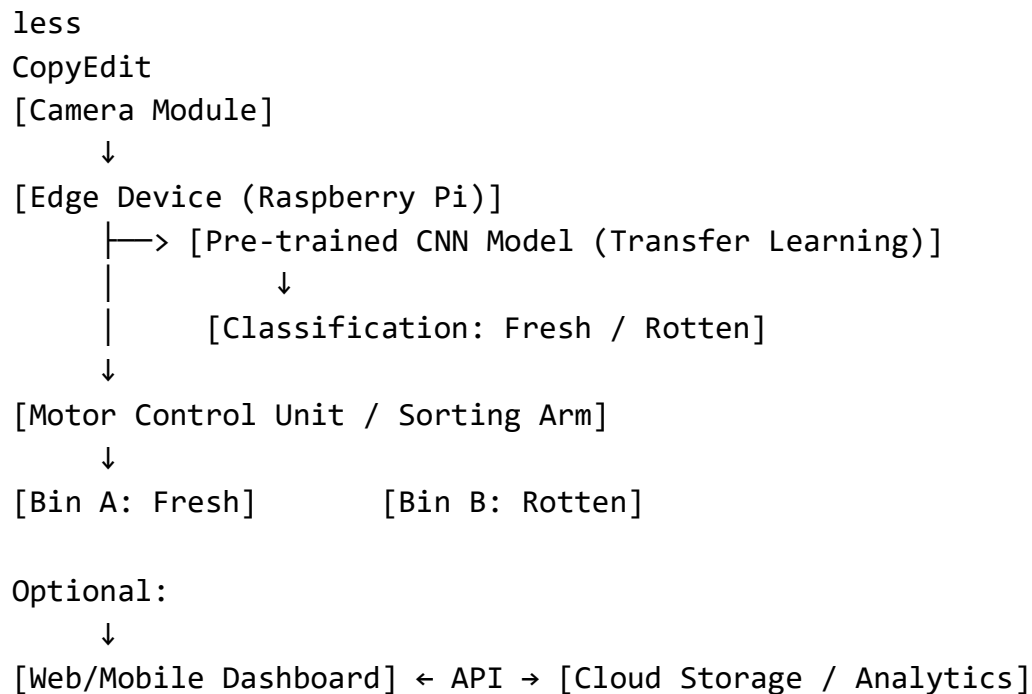
5. User Interface

- a. Optional web/mobile dashboard for visualizing predictions and statistics.
- b. API endpoints for remote monitoring and system control.

6. Cloud (Optional)

- a. For storing historical data, training logs, or re-training models.

Data Flow / Architecture Diagram (Textual Representation)



☒ Features & Development Phases

Phase	Key Activities
Phase 1	Data Collection & Image Labeling
Phase 2	Model Selection, Training & Validation
Phase 3	Hardware Integration (Camera + Raspberry Pi + Motors)
Phase 4	Edge Model Deployment
Phase 5	UI/API Dashboard & Testing
Phase 6	Final Integration and Field Testing