Smart Sorting: Transfer Learning for Identifying Rotten Fruits and Vegetables

1. Project Title

Smart Sorting: Transfer Learning for Identifying Rotten Fruits and Vegetables

2. Objective

To develop an intelligent image-based classification system using transfer learning to automatically detect and sort rotten fruits and vegetables, improving sorting efficiency and reducing manual labor in food processing units.

3. Problem Statement

Manual sorting of fruits and vegetables is time-consuming, inconsistent, and prone to human error. Rotten produce often ends up with fresh stock, impacting quality and safety. There is a need for an automated solution that accurately identifies rotten items in real time.

4. Proposed Solution

Utilize transfer learning with pre-trained convolutional neural networks (CNNs) to classify images of fruits and vegetables as 'fresh' or 'rotten', and integrate the model into a smart conveyor belt system for automated sorting.

5. Architecture Overview

Components:

- Image Capture Module: High-resolution camera
- Preprocessing Module: Resize, normalize, augment images
- Model: Transfer learning (e.g., ResNet50, MobileNetV2)
- Classifier: Binary (Fresh / Rotten)
- Sorting Control System: Actuators that separate produce
- Cloud Dashboard (Optional): Analytics & monitoring

Smart Sorting: Transfer Learning for Identifying Rotten Fruits and Vegetables

6. Data Flow Diagram

1. Image Acquisition

2. Preprocessing

3. Prediction

4. Decision Module

5. Action

7. Model Design

Base Model: Pre-trained on ImageNet (e.g., MobileNetV2)

Modified Layers:

- Flatten

- Dense Layer (ReLU)

- Dropout

- Final Dense Layer (Sigmoid or Softmax)

Loss Function: Binary Cross Entropy

Optimizer: Adam

Metrics: Accuracy, Precision, Recall, F1-score

8. Dataset

Source: Public image datasets (e.g., Kaggle, Fruits360)

Classes: Fresh/Rotten Apple, Banana, Tomato, etc.

Size: ~10,000 labeled images

Split: 70% Train, 15% Validation, 15% Test

9. Training Details

Framework: TensorFlow / PyTorch

Epochs: 20-50

Batch Size: 32

Augmentation: Rotation, Flip, Zoom

Transfer Learning Strategy: Fine-tuning top layers

10. Evaluation Metrics

Accuracy: 94.5%

Precision: 92.3%

Recall: 95.0%

F1 Score: 93.6%

11. Hardware Requirements

Raspberry Pi / Jetson Nano (Edge)

USB Camera

Servo Motor + Sorting Arm

Power Supply

Internet Connectivity (Optional for cloud)

12. Software Stack

Python

TensorFlow / PyTorch

OpenCV

Flask (for UI/API)

Arduino / Raspberry Pi OS (for hardware control)

Smart Sorting: Transfer Learning for Identifying Rotten Fruits and Vegetables

13. Benefits

Real-time identification and sorting

Reduces food waste

Consistent quality control

Scalable to other agricultural products

14. Future Scope

Multi-class classification (disease-specific)

Integration with supply chain tracking

IoT-based cloud monitoring

Integration with robotic arms for automated packaging