SMART SORTING: TRANSFER LEARNING ON ROTTEN FRUITS AND VEGETABLES

Team Lead: SYAM GANGARAJU

Team Members: SAI KIRAN, SAI DURGA PRASAD, MAHIMA

Phase 1: Brainstorming and Ideation

The idea for “Smart Sorting: Transfer Learning on Rotten Fruits and Vegetables” emerged from observing the high

levels of food waste due to inefficient sorting methods in marketplaces and supply chains. During the brainstorming

phase, the team discussed various practical applications of machine learning in agriculture and food industries.

The focus shifted towards a vision system that could automatically classify fruits and vegetables as fresh or rotten,

using transfer learning techniques in deep learning. This approach would leverage pre-trained models to detect rot

with minimal data, increasing efficiency and reducing waste. Discussions were also held about edge implementation

and the feasibility of real-time sorting using computer vision.

Phase 2: Requirements Analysis

In this phase, we defined the technical and functional requirements for the project. The essential components

included:

A dataset of fruits and vegetables with labeled images (fresh vs. rotten)

A computing environment for training the model (Google Colab or local GPU support)

Use of pre-trained CNN models like MobileNetV2, ResNet, or EfficientNet for transfer learning

Performance metrics: Accuracy, Precision, Recall, F1-score

Functional requirements: Detect and classify input images in real-time

Non-functional requirements: Speed, scalability, and low false negatives (especially to avoid letting rotten items

pass as fresh)

Phase 3: Project Design

The design phase involved planning the architecture and flow of the system. The major components of the design included:

Data Pipeline: Image preprocessing, data augmentation, and loading using Keras or PyTorch.

Model Selection: MobileNetV2 was selected for its efficiency and smaller size, making it suitable for edge

deployment.

Training Pipeline: Transfer learning using frozen base layers initially, followed by fine-tuning top layers.

Interface Design: A basic UI for users to upload an image and receive the classification output (fresh/rotten).

Deployment Considerations: Potential deployment via a web application or mobile device, depending on

performance.

Phase 4: Project Planning

The team established a structured timeline and divided responsibilities:

Week 1–2: Dataset collection and preprocessing

Week 3–4: Model training and evaluation

Week 5: UI development and integration

Week 6: Testing and performance optimization

Week 7: Documentation and presentation preparation

Tasks were distributed among members:

Syam (Team Lead): Coordination, Model Training

Sai Kiran: Dataset and Preprocessing

Sai Durga Prasad: Testing and Metrics Analysis

Mahima: UI Development and Documentation

Phase 5: Project Development

The development phase began with sourcing datasets from Kaggle and other open datasets. The images were

cleaned and labeled. Data augmentation techniques such as rotation, zoom, and brightness adjustment were applied

to improve model generalization.

The model was implemented using TensorFlow and Keras. MobileNetV2 was imported with ImageNet weights, with

custom classification layers added. Initial training was conducted on frozen layers, followed by fine-tuning. The best-performing model achieved an accuracy of 93% on the validation set.

A simple web-based UI was created using Flask, enabling users to upload images and receive immediate results on

the classification of the produce.

Phase 6: Functional and Performance Testing

The model underwent extensive testing on different types of fruits and vegetables with varied lighting and

backgrounds. The performance metrics obtained were:

Accuracy: 93.2%

Precision: 92.5%

Recall: 94.1%

F1-Score: 93.3%

Edge cases, such as partially rotten produce, were also tested. The system performed reliably, though performance

decreased slightly in cases with mixed freshness.

User feedback was gathered during testing sessions. Improvements were made in the UI for better usability, and the

model was further fine-tuned based on misclassified samples.

Conclusion

“Smart Sorting: Transfer Learning on Rotten Fruits and Vegetables” presents a cost-effective, scalable solution for

real-time food quality inspection using deep learning. By leveraging transfer learning, the system reduces the need

for extensive training data and delivers high accuracy even with minimal computational resources.

The project demonstrates how AI can contribute to reducing food waste and improving quality control in agriculture

and food logistics.