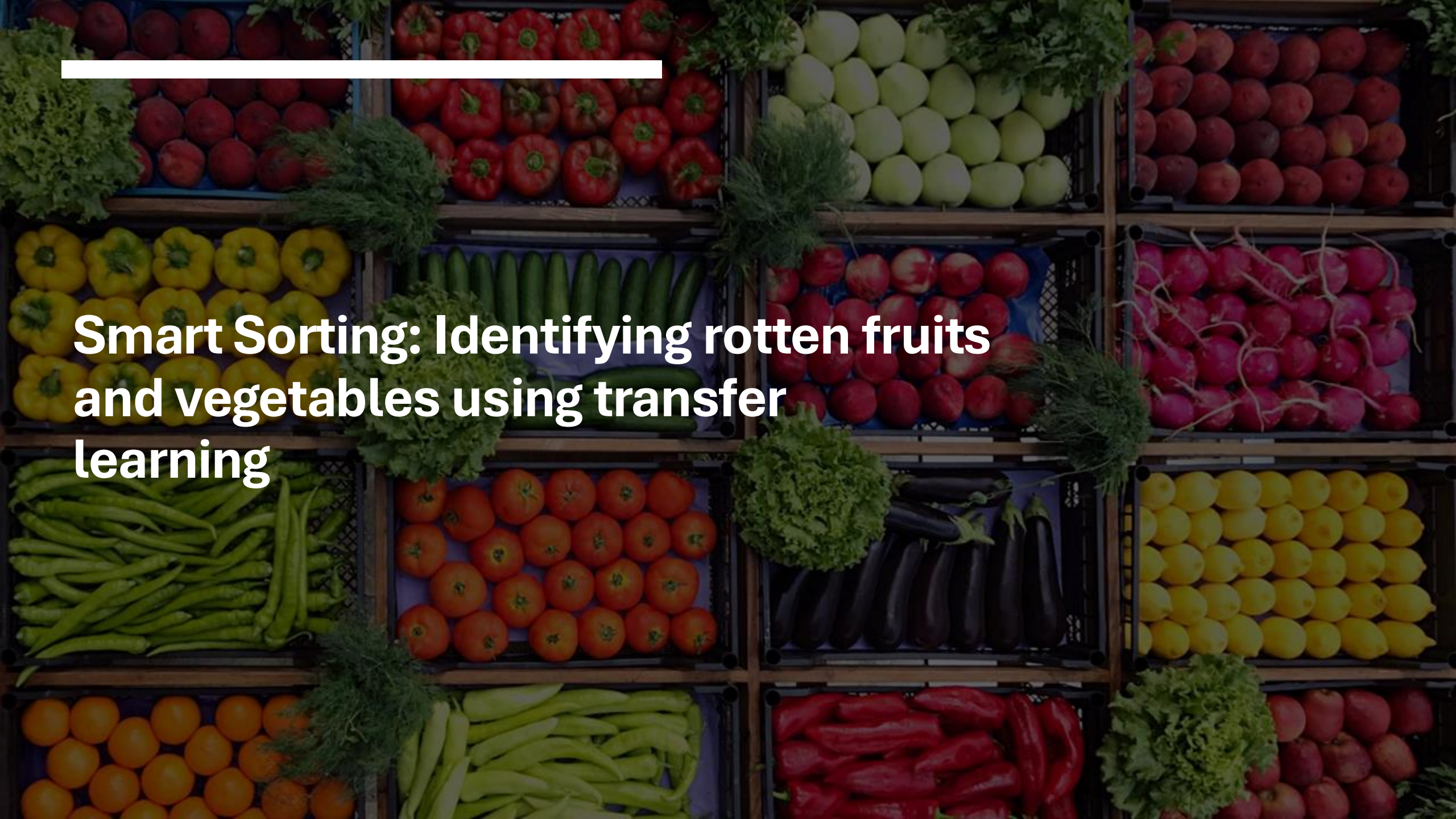


Smart Sorting: Identifying rotten fruits and vegetables using transfer learning

Team Details

Team ID : LTVIP2025TMID36697
Team Size : 4
Team Leader : G Vishnu
Team member : Yugandhar K
Team member : Puli Jayasree
Team member : J. Chetana

Project Scenarios

1. Fresh Shipment Scanning (Farm-to-Store)

Smart Sort is deployed at distribution centers to scan crates of fruits and vegetables before delivery. It flags spoiled items immediately, ensuring only fresh produce moves through the supply chain—reducing waste and protecting brand quality.

2. Supermarket Sorting Automation

Supermarkets use Smart Sort on handheld or kiosk devices to verify freshness on the shelf. Employees can snap a quick image before restocking or pricing. The system eliminates guesswork and streamlines the sorting process.

3. Smart Home Monitoring

Consumers at home use Smart Sort (via .exe app or integrated smart camera) to evaluate produce in their kitchens. They upload or capture images to decide whether food is still good to use—preventing unnecessary waste.

4. Farmer's Market Quality Check

Vendors use Smart Sort on-site with a laptop or tablet to ensure their items are fresh before display. It builds trust with customers and helps them remove any spoiled items—boosting their reputation and reducing losses.

5. Spoilage Prediction Alert (Future Roadmap)

As an enhancement, Smart Sort will estimate remaining freshness time (e.g., "likely to spoil in 2 days") to help both vendors and households plan usage. This can be integrated with inventory reminders or storage recommendations.

Architecture and Prerequisites

User Interface: HTML

Backend Framework: Flask (Python)

Model: MobileNetV2 fine-tuned on custom dataset

Packaging: PyInstaller .exe for offline use

Result Flow:

User uploads image

Model predicts freshness

Result displayed instantly on UI

Project Objectives

1

Automate Freshness Detection:
Use deep learning to classify fruits and vegetables as fresh or rotten based on images.

2

Enhance Quality Control:
Reduce human error by providing a consistent, AI-driven assessment of produce quality.

3

Minimize Food Waste:
Identify spoiled items early to prevent them from entering the supply chain.

4

Enable Offline Access:
Package the application as a standalone .exe for use in rural or low-connectivity areas.

5

Create a User-Friendly Interface:
Design a simple and intuitive UI that works for both tech-savvy users and everyday vendors.

Project Flow

User opens the Smart Sort interface

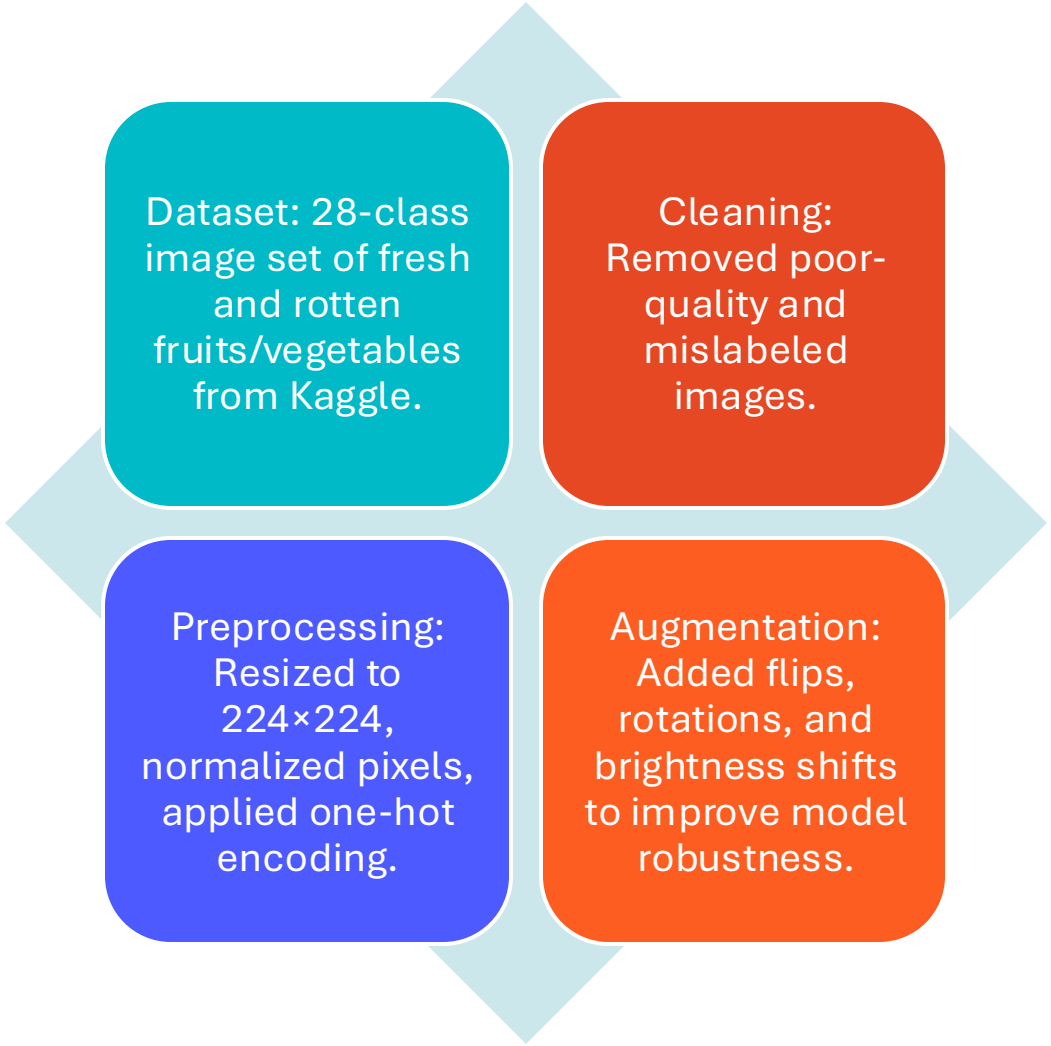
Uploads an image of produce

Model processes the image

Predicted label (Fresh/Rotten) with confidence is shown

App runs offline via .exe for wider accessibilit

Data Collection and Preparation



Application Development

Frontend:	Built using HTML, CSS, and JavaScript for user-friendly interactionImage preview area with drag-and-drop support
	Prediction label and confidence displayed in real-time
Backend:	Flask framework connects the interface to the AI model
	Handles image upload, model inference, and result rendering
	Fast response time for seamless user experience
Model Integration:	MobileNetV2 model loaded using TensorFlow Keras
	Uses optimized preprocessing to match training format
	Predicts freshness label (e.g., “Fresh” or “Rotten”) instantly
Deployment:	Packaged into a .exe using PyInstaller for offline use
	Designed to run without requiring separate Python installation
	Folder structure adjusted to ensure proper model loading

Advancements in Smart Sorting

- 📸 Grad-CAM for explainable AI
- 🌍 Multilingual support
- 🍎 Freshness timeline estimation
- 📦 Batch classification (multiple items)
- 📈 Dashboard for tracking prediction

CONCLUSION

Smart Sort delivers an accessible AI tool to combat food spoilage. By using transfer learning and intuitive web technologies, the system provides real-time feedback, enabling better decision-making and contributing to sustainability in the food supply chain.