FRUIT FRESHNESS CLASSIFIER

IMAGE - BASED CLASSIFICATION OF FRUITS

Presented By: Vaibhav Garg

PROBLEM STATEMENT

- Identifying the freshness of fruits from images is often unreliable because of differences in lighting, camera quality, viewing angles, and fruit variety. These inconsistencies make visual inspection difficult to standardize.

- Manual inspections in markets, warehouses, and supply chains can be slow, labor-intensive, and influenced by human bias, sometimes resulting in inaccurate evaluations and avoidable waste.

- An efficient, technology-driven system is needed to assess fruit quality quickly and consistently, regardless of environmental variations.

- By applying deep learning techniques, freshness classification can be automated, delivering fast, accurate, and repeatable results that improve quality control and streamline sorting processes.

PROJECT OBJECTIVES

- Automatically classify fruit images into one of 16 categories: eight fresh fruit classes and their corresponding stale counterparts.
- Achieve more than 95% accuracy while training the model in three or fewer epochs on a validation dataset of labeled fruit images captured under varied lighting and background conditions.
- Provide an interactive Streamlit web app where users can upload a fruit image and instantly receive a freshness prediction.
- Enhance quality control in supply chains by delivering consistent, objective, and evidence-based freshness assessments from uploaded images.

DATASET OVERVIEW

- The dataset contains 16,000 labeled fruit images, designed to cover a diverse range of fruit types and freshness conditions.

- Images are evenly distributed across 16 categories — eight types of fresh fruits and their stale counterparts — ensuring balanced representation for model training:
  - **Fresh Fruits**: Banana, Lemon, Lulo, Mango, Orange, Strawberry, Tamarillo, Tomato.
  - **Stale Fruits:** Banana, Lemon, Lulo, Mango, Orange, Strawberry, Tamarillo, Tomato.

- Each category contains 1,000 images, captured under varied lighting, angles, and backgrounds to simulate real-world market and storage conditions.

- The dataset's diversity helps the model generalize better to unseen images, making it robust for deployment in quality control and retail applications.

DATA PREPROCESSING

- **Image Transformations (Data Augmentation & Standardization)**
  - **RandomHorizontalFlip()** – Flips images horizontally to account for variations in fruit orientation.
  - **RandomRotation(10)** – Rotates images randomly up to ±10° to simulate different camera angles.
  - **ColorJitter(brightness=0.2, contrast=0.2)** – Randomly adjusts brightness and contrast to mimic diverse lighting conditions.
  - **Resize((224, 224))** – Resizes all images to 224×224 pixels, matching the ResNet50 input requirements.
  - **ToTensor()** – Converts images to PyTorch tensors, scaling pixel values to the [0,1] range.
  - **Normalize(mean=[0.485, 0.456, 0.406], std=[0.229, 0.224, 0.225])** – Standardizes images using ImageNet statistics for compatibility with pre-trained CNN weights.

- **Data Splitting**

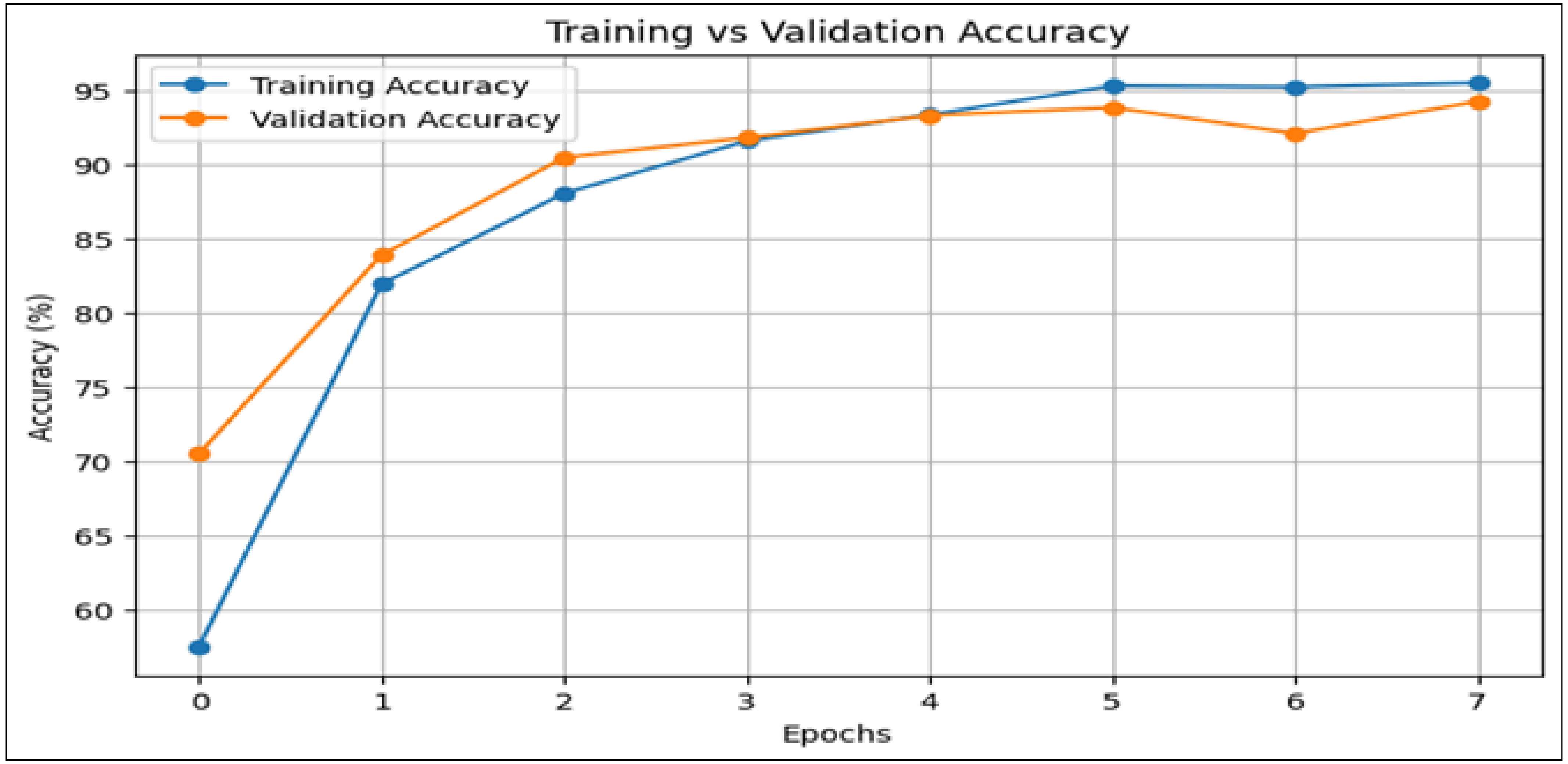
  - **Training Set (70%)** – 11,200 images used for learning model weights.
  - **Validation Set (15%)** – 2,400 images used to monitor model performance and detect overfitting.
  - **Test Set (15%)** – 2,400 images reserved for final model evaluation.

- **Data Loading**

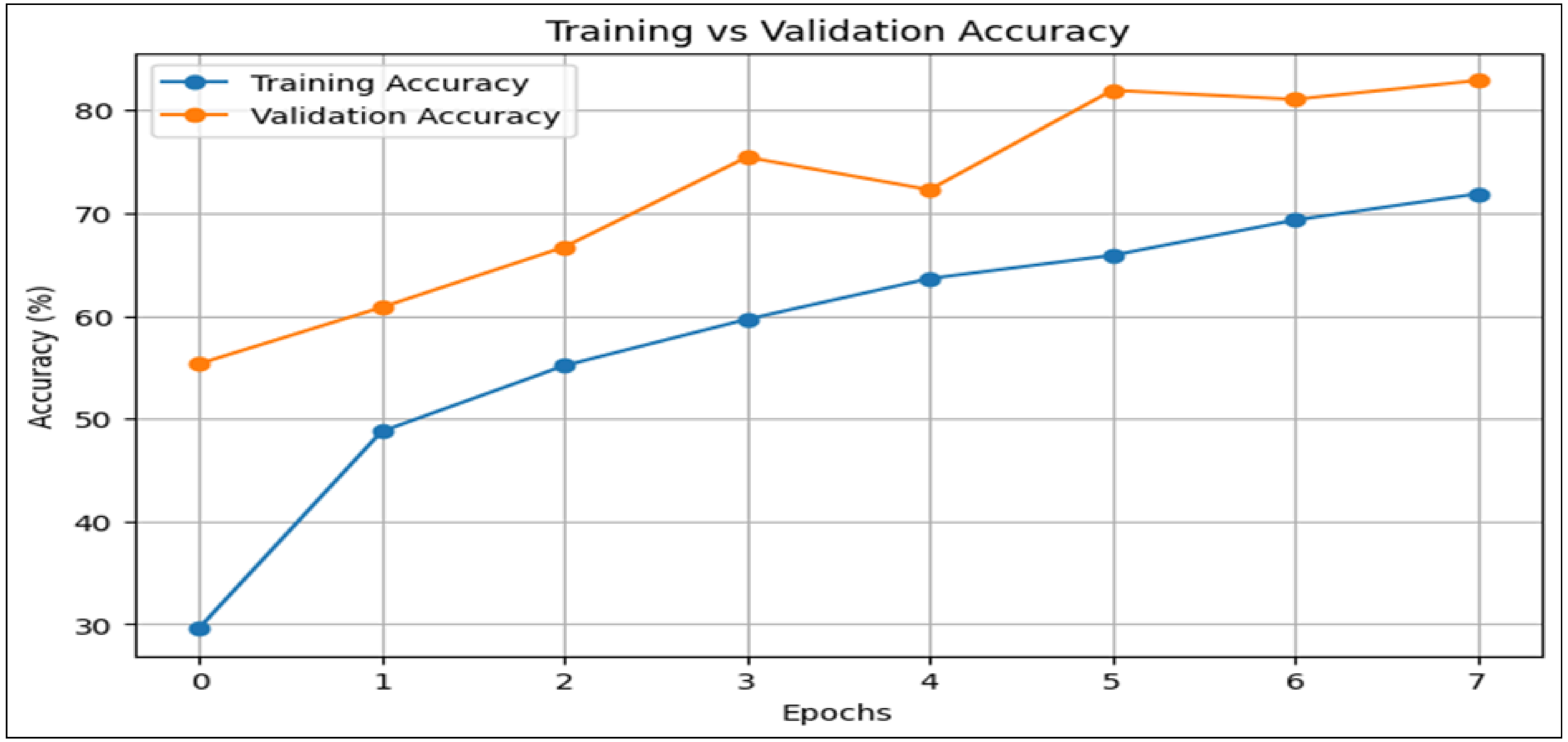
  - **Training Loader** – Batch size: 32, data shuffled for better generalization.
  - **Validation & Test Loaders** – Batch size: 32, shuffling disabled to preserve evaluation order.
  - **Multi-process Loading** – Accelerates batch preparation during training and validation phases.

MODEL TRAINING

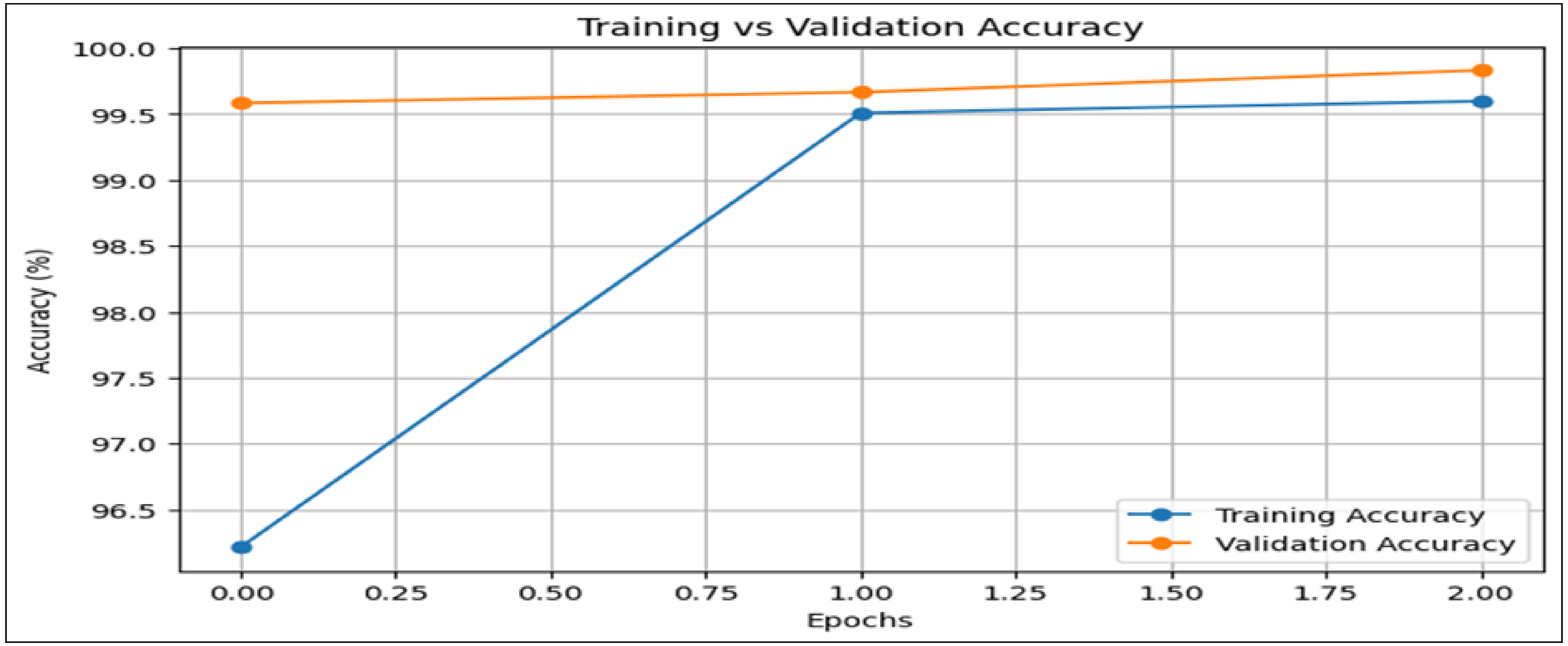
- **Model 1 – CNN (Baseline)**
  - **Architecture:** Standard convolutional neural network without explicit regularization.
  - **Epochs:** 8
  - **Performance:** Train Acc = 95.51%, Val Acc = 94.25%, Test Acc = 92.92%.
  - **Observation:** High accuracy but slight overfitting observed after ~5 epochs.



- **Model 2 – CNN with Regularization**
  - **Architecture:** CNN with added BatchNorm2d, Dropout (p = 0.3), and L2 Regularization (weight decay = 1e-5).
  - **Epochs:** 8
  - **Performance:** Train Acc = 71.89%, Val Acc = 82.92%, Test Acc = 83.42%.
  - **Observation:** Reduced overfitting compared to Model 1 but lower overall accuracy due to stronger regularization.



- **Model 3 – Transfer Learning (ResNet50)**
  - **Architecture:** Pretrained ResNet50 with fine-tuning of final layers.
  - **Epochs:** 3
  - **Performance:** Train Acc = 99.60%, Val Acc = 99.83%, Test Acc = 99.71%.
  - **Observation:** Achieved exceptional accuracy within 3 epochs, minimal overfitting, fastest convergence.

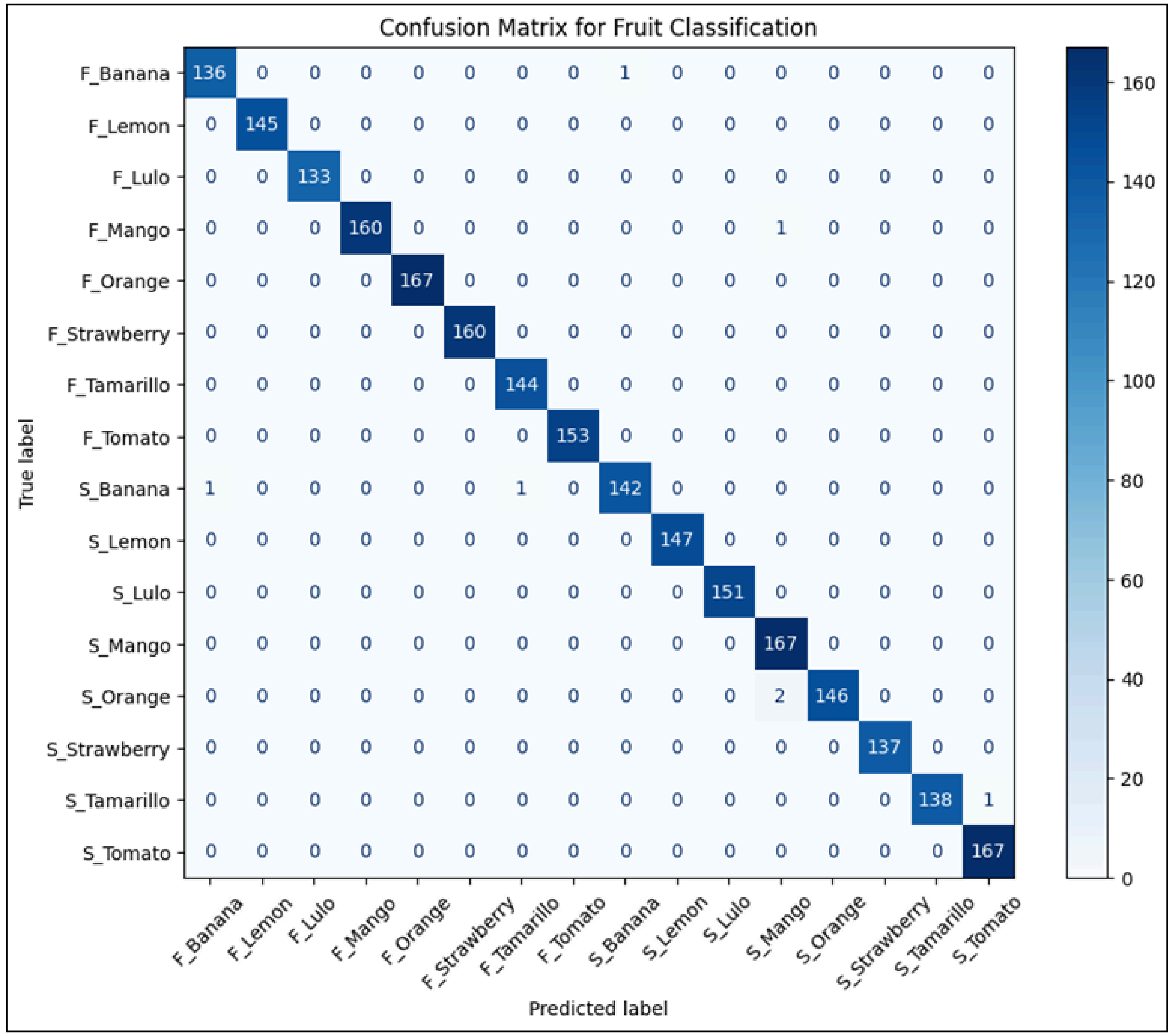


MODEL EVALUATION

- **Classification Report**
  - **Metrics Generated:** Precision, recall, and F1-score calculated for all 16 classes.
  - **Performance**: Achieved ~100% accuracy, with macro and weighted averages both at 1.00.
  - **Class-Level Insights:** Minimal variation in performance — most classes scored 1.00 in all metrics.
  - **Objective Check:** Far exceeded target of >95% accuracy after tuning.

	precision	recall	f1-score	support
0	0.99	0.99	0.99	137
1	1.00	1.00	1.00	145
2	1.00	1.00	1.00	133
3	1.00	0.99	1.00	161
4	1.00	1.00	1.00	167
5	1.00	1.00	1.00	160
6	0.99	1.00	1.00	144
7	1.00	1.00	1.00	153
8	0.99	0.99	0.99	144
9	1.00	1.00	1.00	147
10	1.00	1.00	1.00	151
11	0.98	1.00	0.99	167
12	1.00	0.99	0.99	148
13	1.00	1.00	1.00	137
14	1.00	0.99	1.00	139
15	0.99	1.00	1.00	167
accuracy			1.00	2400
macro avg	1.00	1.00	1.00	2400
weighted avg	1.00	1.00	1.00	2400

- **Confusion Matrix**
  - **Visualization:** Plotted actual vs. predicted labels for all classes.
  - **Result:** Almost perfect diagonal matrix, indicating excellent classification capability.
  - **Misclassifications:** Very few cases like minor confusion between **F_Banana** & **S_Banana** and **F_Mango** & **S_Mango**.
  - **Interpretation:** Model generalizes exceptionally well across different fruit types and states (**fresh/stale**).



STREAMLIT APP INTEGRATION

- Built an interactive **Streamlit** app for real-time fruit classification, integrating the trained **ResNet50** model.

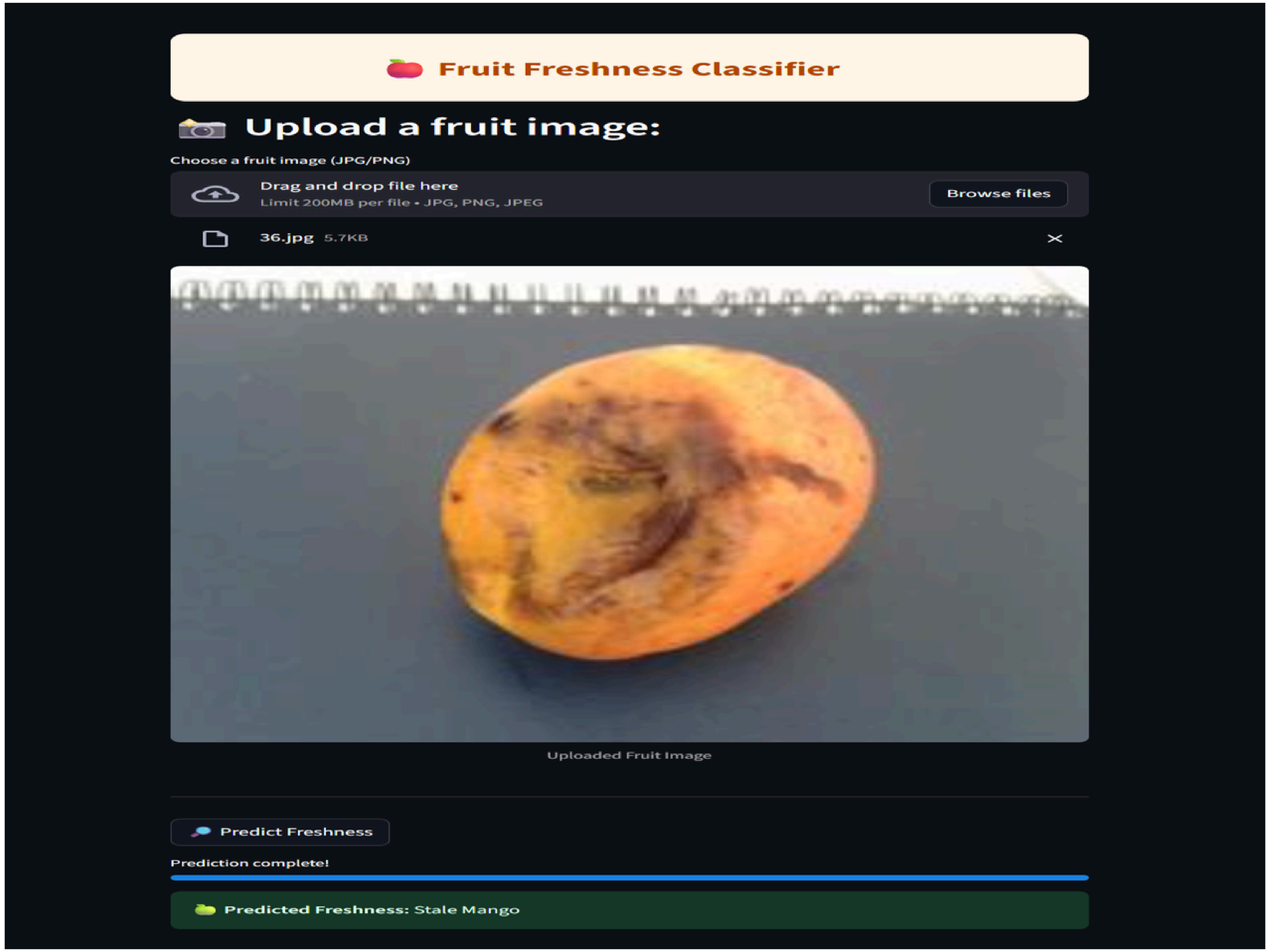
- Includes in-app preprocessing (resizing, normalization) for smooth and accurate predictions.

- Users can upload a clear, centered fruit image via a simple, user-friendly interface.

- On clicking "**Predict Fruit Class**", the app displays the predicted category (e.g., Fresh Banana, Spoiled Mango) alongside the uploaded image.

- Deployed publicly on **Streamlit Cloud** for easy access, enabling quick, objective, and repeatable assessments for businesses and quality inspectors.

USER INTERACTION PREVIEW



PROJECT SUMMARY

- Built a deep learning system to classify fruits as Fresh or Spoiled across eight fruit types, totaling 16 categories (e.g., Fresh Banana, Spoiled Mango).
- Prepared a dataset of 16,000 labeled images, applying augmentations such as horizontal flips, rotations, and brightness/contrast adjustments for better generalization.
- Split the dataset into 70% training, 15% validation, and 15% test sets, ensuring balanced class representation for reliable evaluation.
- Trained baseline CNN and ResNet50 models with batch normalization, dropout (0.3), and weight decay (1e-5); tuned ResNet50 to achieve near-perfect accuracy within ≤3 epochs.
- Evaluated performance using classification reports and confusion matrices, achieving ~100% accuracy and identifying minimal misclassification cases.
- Developed and deployed an interactive Streamlit app for real-time fruit freshness classification.
- Live App: https://vaibhav-project-fruit-freshness-classifier.streamlit.app/
- GitHub: https://github.com/vaibhavgarg2004/Fruit-Freshness-Classifier

THANK YOU