**AI Object Recognition Project on Raspberry Pi**

**Day 1: Project Planning and Setup**

* Defined project goals and scope.
* Researched AI models suitable for object recognition on Raspberry Pi.
* Selected YOLO v11 for real-time image processing.
* Installed necessary software and dependencies on the Raspberry Pi.

**Day 2: Hardware Setup**

* Set up the Raspberry Pi with a camera module.
* Verified camera functionality using test scripts.
* Ensured stable power supply and network connectivity.

**Day 3: Dataset Collection**

* Researched publicly available image datasets for food item recognition.
* Downloaded and preprocessed images (resizing, normalization, labeling).
* Organized dataset into training and validation sets.

**Day 4: Model Selection and Initial Training**

* Installed Ultralytics YOLO framework.
* Selected a pre-trained YOLO v11 model for transfer learning.
* Configured training parameters (learning rate, batch size, number of epochs).

**Day 5: Training the Model**

* Started model training on a high-performance computer.
* Monitored training progress and loss function convergence.
* Saved model checkpoints for future reference.

**Day 6: Evaluating Model Performance**

* Validated model using test data.
* Analyzed accuracy, precision, recall, and F1-score.
* Identified misclassified images and considered dataset augmentation.

**Day 7: Model Optimization for Raspberry Pi**

* Converted model to a lightweight format (NCNN).
* Quantized model for efficient performance on Raspberry Pi.
* Tested inference speed and adjusted parameters accordingly.

**Day 8: Deployment on Raspberry Pi**

* Transferred optimized model to Raspberry Pi.
* Wrote a script to capture images and run inference using YOLO.
* Ensured real-time processing capabilities.

**Day 9: Initial Testing and Debugging**

* Conducted real-world tests with different food items.
* Debugged issues related to incorrect detections.
* Adjusted confidence thresholds and fine-tuned parameters.

**Day 10: Brand Recognition Implementation**

* Collected brand logos for training.
* Fine-tuned the model to differentiate between brands.
* Tested model accuracy on branded food packaging.

**Day 11: Calorie and Nutrition Information Integration**

* Created a database of food items with calories and nutrition details.
* Linked detected objects to their corresponding nutrition data.
* Displayed relevant information on the Raspberry Pi screen.

**Day 12: User Interface Development**

* Designed a simple UI to display detected items and nutritional facts.
* Implemented voice feedback using text-to-speech (optional feature).
* Optimized UI responsiveness for real-time updates.

**Day 13: System Testing and Performance Evaluation**

* Conducted extensive testing with various food items.
* Measured inference time and accuracy under different lighting conditions.
* Logged results for future improvements.

**Day 14: Final Optimization and Documentation**

* Tweaked model parameters for better accuracy.
* Optimized power consumption on Raspberry Pi.
* Documented the entire development process for future reference.

**Day 15: Project Review and Future Improvements**

* Reviewed overall project performance.
* Identified areas for further improvement (e.g., dataset expansion, model enhancements).
* Prepared a final report summarizing the findings and results.