Project Report: Object Detection & Model Maintenance with Falcon

Title Page

Team Name: Debug Thugs

Project Name: AI Engineering & Object Detection Optimization

Brief Tagline: *Empowering Vision through Intelligent Maintenance and Real-Time Detection.*

Methodology

Steps Taken While Training the Model

This section details the pipeline used to train and fine-tune the object detection model.

1. **Data Collection & Preprocessing:**
   - Describe the initial dataset used.
   - Mention any preprocessing steps (resizing, normalization, labeling).
2. **Model Selection:**
   - Specify the base model used (e.g., YOLO, Faster R-CNN) and why it was chosen.
3. **Training Process:**
   - **Hyperparameters:** (Batch size, learning rate, epochs).
   - **Platform:** Mention the use of the Falcon account/platform for training or management.
4. **Fine-Tuning Results:**
   - Describe how the model was fine-tuned on specific classes.
   - *Example:* "Initial training showed low accuracy on small objects, so we adjusted the anchor box sizes and retrained."

Results & Performance Metrics

Performance Analysis

Objective metrics demonstrating the model's accuracy.

- **mAP Scores (Mean Average Precision):**
  - **Baseline Model:** [Insert Score, e.g., 40%]
  - **Final Model:** [Insert Score, e.g., 55%]
  - *Visual:* [Insert Bar Chart comparing mAP scores]
- **Confusion Matrix:**
  - *Visual:* [Insert Confusion Matrix Image here]
  - *Analysis:* "The matrix shows that the model successfully distinguishes between Class A and Class B, with minimal false positives."
- **Accuracy Comparisons:**
  - Compare the model's performance before and after specific interventions (like data augmentation).

Failure Cases (Problem -> Fix -> Result)

As per instructions, include images of failure cases and how they were fixed.

- **Failure Case 1: Occlusion**
  - **Problem:** The model failed to detect objects partially hidden behind others.
  - **Image:** [Insert image of failed detection]
  - **Solution:** We augmented the dataset with artificially occluded examples.
  - **Result:** New mAP Score improved by X%.

Challenges & Solutions

Key Obstacles and Resolutions

Challenge	Impact	Solution Implemented
Limited Dataset Diversity	The model performed poorly in low-light conditions.	**Data Augmentation:** Applied brightness/contrast adjustments to training data to simulate night conditions.
Overfitting	High training accuracy but low validation accuracy.	**Regularization:** Implemented dropout layers and early stopping during training.
Integration with Falcon	Initial difficulty in configuring the Falcon pipeline.	**Documentation & Iteration:** Consulted Falcon docs and refactored the data ingestion script to match Falcon's requirements.
Real-time Latency	Inference time was too slow for video feeds.	**Model Quantization:** Converted the model to a lighter format (e.g., ONNX/TFLite) to improve FPS.

Conclusion & Future Work

Final Thoughts

The "Debug Thugs" team successfully engineered a robust object detection pipeline. By leveraging the Falcon platform, we were not only able to train the model but also establish a workflow for continuous improvement. The final model achieves a [Insert final mAP]% mAP, making it suitable for [Insert target use case].

Potential Improvements & Future Work

1. **Edge Deployment:** Porting the model to run entirely on edge devices (Raspberry Pi/Jetson Nano) without cloud dependency.
2. **Advanced Augmentation:** Using Generative AI to create synthetic training data for rare edge cases.
3. **Cross-Domain Adaptation:** Testing the model in completely different environments (e.g., aerial view vs. street view) to ensure robustness.

Use Case Proposal

1. Application Concept

App Name: [Insert App Name, e.g., "SafeSight"] **Platform:** [Computer/Phone] **Real-World Problem:**

- Describe a specific problem (e.g., safety gear detection in construction zones).
- *Solution:* The application monitors video feeds in real-time and alerts supervisors if a worker is missing a helmet or vest.

2. Maintaining the Model with Falcon

Continuous Learning & Adaptation:

- **Scenario:** If a new type of safety helmet (e.g., a different color) is introduced, the current model might fail.
- **Falcon's Role:**
  1. **Data Drift Detection:** Falcon monitors the confidence scores. If the model consistently has low confidence on the new helmets, Falcon flags these images.
  2. **Active Learning:** These "low confidence" images are sent for human verification/labeling.
  3. **Retraining:** Falcon automatically triggers a retraining pipeline including these new examples, updates the model weights, and redeploys the improved version to the application.