Efficient Object Detection and Robustness Analysis

Overview

This project presents a complete object detection pipeline built on a small Pascal VOC subset using transfer learning and deployment-focused optimization. The objective was not only to train an accurate detector but also to evaluate generalization, analyze robustness, and improve inference efficiency under real-world constraints.

The dataset consists of three classes: **person, car, and dog**. Due to the limited data size, the emphasis was placed on controlled generalization, efficient training, and deployment readiness rather than large-scale model scaling.

Model Training and Architecture

A pretrained **YOLOv8** detector was selected for this task because of its strong balance between accuracy and real-time performance. YOLOv8 uses an anchor-free detection head and an optimized training pipeline, enabling faster convergence and improved stability on smaller datasets compared to older detectors.

Transfer learning was applied by fine-tuning pretrained weights on the Pascal VOC subset. This approach allows the model to reuse rich visual features learned from large-scale datasets while adapting to the target classes.

The model utilizes YOLOv8's default **CSP-based backbone with multi-scale feature aggregation**, which provides robust feature extraction across object scales without significantly increasing computational complexity.

Final Performance

- mAP@0.5 ≈ 0.84
- Precision ≈ 0.86
- Recall ≈ 0.75

The model converged smoothly and produced consistent multi-class detection results, demonstrating strong transfer learning effectiveness even with limited data.

Overfitting and Generalization Analysis

Given the small dataset size, overfitting was a major concern. Training and validation curves were analyzed to assess generalization behavior.

The training and validation losses followed similar trends throughout the training process, with no major divergence observed. While training loss decreased slightly faster, validation performance remained stable, indicating that the model did not overfit severely.

Several measures helped maintain generalization:

- Transfer learning from pretrained weights
- Built-in augmentation from the YOLO training pipeline
- Controlled training duration to avoid memorization
- Implicit regularization through batch normalization

Together, these techniques ensured stable validation performance despite the limited dataset size.

Latency Optimization for Deployment

Beyond accuracy, the model was optimized for real-world inference efficiency. Baseline inference was measured using an input resolution of 768 pixels, which provided strong accuracy but higher computational cost.

To improve latency, the input resolution was reduced from **768 to 512 pixels**. This significantly lowers computational complexity while preserving most semantic information.

Performance Impact

- FPS before optimization: ~57 FPS
- FPS after optimization: ~96 FPS
- Latency improvement: ~69%
- Accuracy drop: minimal (~0.014 mAP)

This demonstrates a practical speed–accuracy trade-off suitable for real-time deployment scenarios.

To further enhance deployment readiness, the model was exported to **ONNX format**, enabling cross-platform compatibility. Additionally, **dynamic INT8 quantization** was applied to reduce model size and improve CPU inference efficiency. These optimizations make the model suitable for edge devices and resource-constrained environments.

Failure Analysis and Robustness

To better understand real-world limitations, incorrect predictions were manually analyzed. Several recurring failure modes were identified.

Key Failure Patterns

1. Missed detections for very small objects
2. Partial occlusion reducing visible features
3. Overlapping objects in cluttered scenes
4. Low contrast between object and background
5. Rare poses not represented in training data

Most errors appear to be **data-driven rather than architectural**, indicating that the model's limitations stem primarily from insufficient training diversity rather than model capacity.

Potential Improvements

- Multi-scale training to improve small-object sensitivity
- Larger and more diverse datasets
- Hard example mining strategies
- Higher-resolution inference for edge cases
- Test-time augmentation for improved robustness

This analysis highlights realistic deployment challenges and demonstrates an understanding of model behavior beyond standard metrics.

Conclusion

This work delivers a deployment-aware object detection pipeline that balances accuracy, efficiency, and robustness. By combining transfer learning, careful generalization analysis, and inference optimization, the model achieves strong performance under practical constraints.

Key contributions include:

- Reliable detection performance on limited data
- Effective overfitting control through transfer learning and augmentation
- Significant latency improvements via resolution scaling
- Deployment readiness through ONNX export and quantization
- Realistic failure analysis highlighting practical limitations

Overall, the project reflects a holistic understanding of modern computer vision systems, focusing not only on model accuracy but also on **generalization, efficiency, and real-world applicability**.