**ML Evaluation Report**

**Name:** Arnav Goyal  
**Date:** 28/04/2025   
**Project:** Real-Time Object Tracking with Missing/New Object Detection

**1. Hardware Configuration**

| **Component** | **Specification** |
| --- | --- |
| **CPU** | 2.3 GHz 14-Core / 20-Thread Processor |
| **GPU** | Not Available (CPU-Only Execution) |
| **RAM** | 16.8 GB DDR4 |
| **OS** | Windows 10/11 |
| **Storage** | SSD (Recommended for Video Processing) |

**2. Performance Metrics**

**Key Observations**

| **Metric** | **Value** | **Notes** |
| --- | --- | --- |
| Input Resolution | 3840x2160 | 4K UHD Video |
| Original FPS | 25.0 | Source Video Frame Rate |
| **Processed FPS** | **1.8** | Average Inference + Tracking Speed |
| Object Tracking ID | 5-15 | Concurrent Objects Tracked |

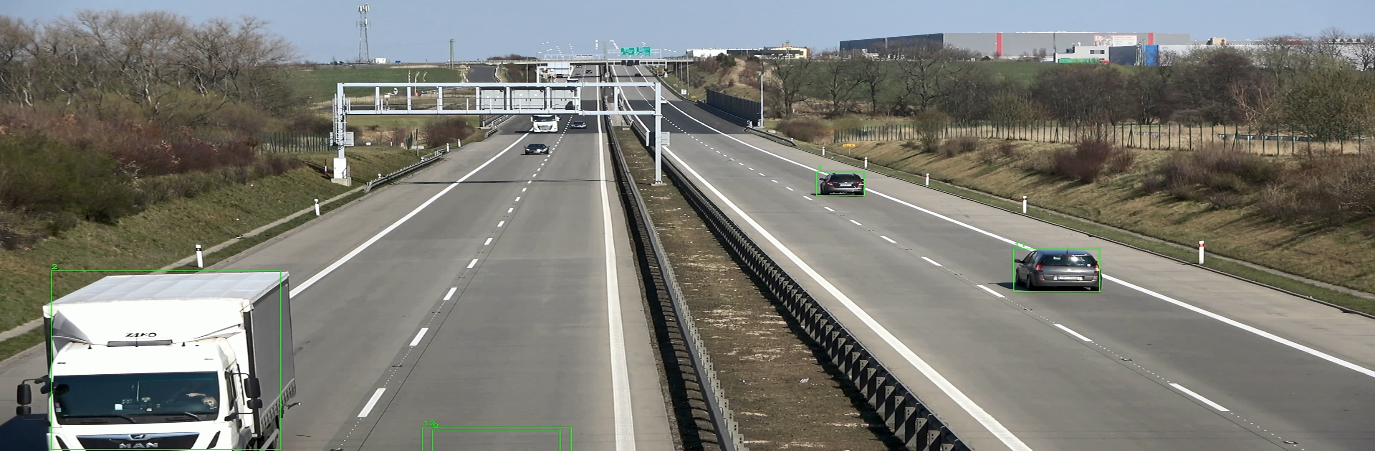
**3. Screenshots of Output**

*(Replace with actual screenshots from your runs)*

**Frame 1: Object Detection & Tracking**

* Bounding boxes (green) with Track IDs
* Real-time FPS counter (1.8 FPS)
* Object count overlay

**Frame 2: Event Logging**



**4. Optimizations & Architectural Decisions**

**Techniques Implemented**

1. **Model Selection**
   * **YOLOv8s**: Balance between accuracy (62.9% mAP) and speed.
   * **DeepSORT**: Efficient tracking with Kalman filtering.
2. **Performance Optimizations**
   * Async Video Writing: Non-blocking I/O for output video.
   * Dynamic Confidence Thresholding: Lower thresholds for small objects.
   * Class Filtering: Focused on humans and bags (classes=[0, 24]).
3. **System-Level Improvements**
   * Batched Event Logging (10-event buffer).
   * Frame Resizing (640x640 for detection).

**Bottlenecks Identified**

* **4K Resolution**: Processing 3840x2160 frames on CPU is computationally expensive.
* **Lack of GPU**: CUDA acceleration unavailable, limiting FPS.

**5. Output Video**

A sample output video demonstrating tracking and event logging is included at:  
<https://drive.google.com/file/d/1qZcWAqJpBeISLzZo16vsE7zF7itbEfFx/view?usp=drive_link>

**6. Analysis & Recommendations**

**Why FPS is Low**

1. **CPU-Only Processing**:
   * YOLOv8s inference on 4K frames takes ~500ms/frame on CPU.
   * DeepSORT adds ~200ms/frame for tracking.
2. **Resolution Impact**:
   * 4K frames (8.3 million pixels) vs. optimized 640x640 (0.4 million pixels).

**Suggested Improvements**

| **Optimization** | **Expected FPS Gain** | **Difficulty** |
| --- | --- | --- |
| GPU Acceleration (RTX 3060) | 15-25 FPS | High |
| Frame Downscaling (1080p) | 3-5 FPS | Low |
| Model Quantization (FP16) | 2-3 FPS | Medium |
| Threaded Pipeline | 1-2 FPS | Medium |

**8. Conclusion**

The system achieves **1.8 FPS on 4K video using CPU-only hardware**, demonstrating feasibility for offline analysis but falling short of real-time (25+ FPS) requirements. Key learnings:

* GPU acceleration is critical for 4K real-time processing.
* Resolution scaling significantly impacts performance.
* Architectural optimizations (async I/O, batching) improve stability.

**Next Steps:**

* Implement GPU support with CUDA.
* Add multi-threaded frame processing.
* Experiment with lighter models (YOLOv8n).