EQUINOX

MILITARY OBJECT DETECTION SYSTEM: TECHNICAL REPORT

EXECUTIVE SUMMARY

Current aerial reconnaissance AI suffers from a critical flaw: it is excellent at detecting large, common objects (e.g., tanks in open terrain) but fails catastrophically at identifying high-value, rare assets such as hidden trenches or camouflaged soldiers.
Project Equinox is a Dual-Core Intelligence System designed to bridge this gap. By combining Synthetic Data Engineering with a novel Ensemble Inference Architecture, we deliver:

- **Maximum Recall** – Rare anomalies are detected instead of ignored
- **High Precision** – False alarms are aggressively filtered
- **Real-Time Performance** – Operating at 62 FPS on standard hardware using OpenVINO

PROBLEM ANALYSIS & DATASET CHALLENGES

Why This Was Hard
The dataset presented a realistic yet extreme challenge for standard AI pipelines.

2.1 Extreme Class Imbalance

- Tanks / Vehicles: ~7,000+ samples
- Trenches: 2 samples
- Civilians: 18 samples

Challenge:
Conventional models learn that rare classes "do not exist" and suppress them entirely.

2.2 The "Ant Perspective" (Tiny Objects)

- Objects like soldiers or missile launchers occupied <1% of total pixels
- *Downscaling 4K images to* 640p destroys these features entirely

2.3 Camouflage & Occlusion

- Targets hidden by trees, smoke, shadows
- Visual similarity to background breaks edge-based detection

METHODOLOGY

Pillar 1: Synthetic Data Injection (Solving Imbalance)

Since real data collection was impossible, we manufactured data.

- ***Technique****: Copy-Paste Augmentation with Poisson Blending*
- ***Process****:*
  -Extracted scarce objects (trenches, civilians)
  -Rotated, scaled, and context-matched them
  -Seamlessly injected into thousands of new environments
- ***Outcome****:*
  -Converted a 2-shot learning problem into a 500+ sample dataset
  -The model learned trenches as a real class, not noise

Pillar 2: Dual-Core Training Strategy

One model was not enough. We trained two specialized intelligences.

- Gold Model
  - Trained for 145 epochs
  - High precision, low false positives
  - Acts as a sniper — fires only on confidence
- Nuclear Model
  - YOLOv8-ExtraLarge
  - High recall, anomaly-sensitive
  - Acts as a spotter — misses nothing suspicious

Pillar 3: Edge Optimization

- Models quantized to OpenVINO INT8
- Achieved:
  - 4× smaller model size
  - 300% speed increase
- Suitable for drone-grade hardware deployment

4. Model Architecture Selection

YOLOv8 was selected over R-CNNs and Transformers due to:

1. Speed vs Accuracy - Only architecture capable of >50 FPS, mandatory for live drone feeds
2. Mosaic Augmentation -Learns contextual relationships (e.g., soldiers near tanks)
3. SAHI Compatibility-Perfect integration for tiled high-resolution inference

Selected Variants

- YOLOv8-Large (Gold Model)- Balance of precision, generalization, and stability
- YOLOv8-X (Nuclear Model)- Brute-force detection for extreme edge cases

TRAINING HARDWARE

- All model training was performed on an ASUS TUF-15 laptop equipped with an NVIDIA GeForce RTX 4090 GPU.

Inference Strategy: Dual-Core Ensemble

Instead of a single inference path, Equinox uses parallel intelligence streams.

Pipeline

- Input: Satellite / drone imagery
- Parallel Processing:
  -Gold Model: High-confidence detections
  -Nuclear Model: Hidden / obscure detections
- Fusion (NMS):
  -Gold → Tree
  -Nuclear → Soldier in Tree
  -Output → Soldier flagged
- SAHI Slicing: High-resolution images tiled to preserve micro-objects

RESULT : Behaves like multiple human analysts working in parallel.

RESULTS & PERFORMANCE

"Proven Metrics"

- ***mAP@50: ~70% -*** (Other Standard Baseline Models: 40–45%, Dual-Core gain: +5%)
- ***Real-World Reliability: ~85%*** -(4/5 targets detected, including camouflaged & occluded)
- ***Inference Speed: 62.31 FPS-*** (CPU-verified using Intel OpenVINO; multi-feed capable)

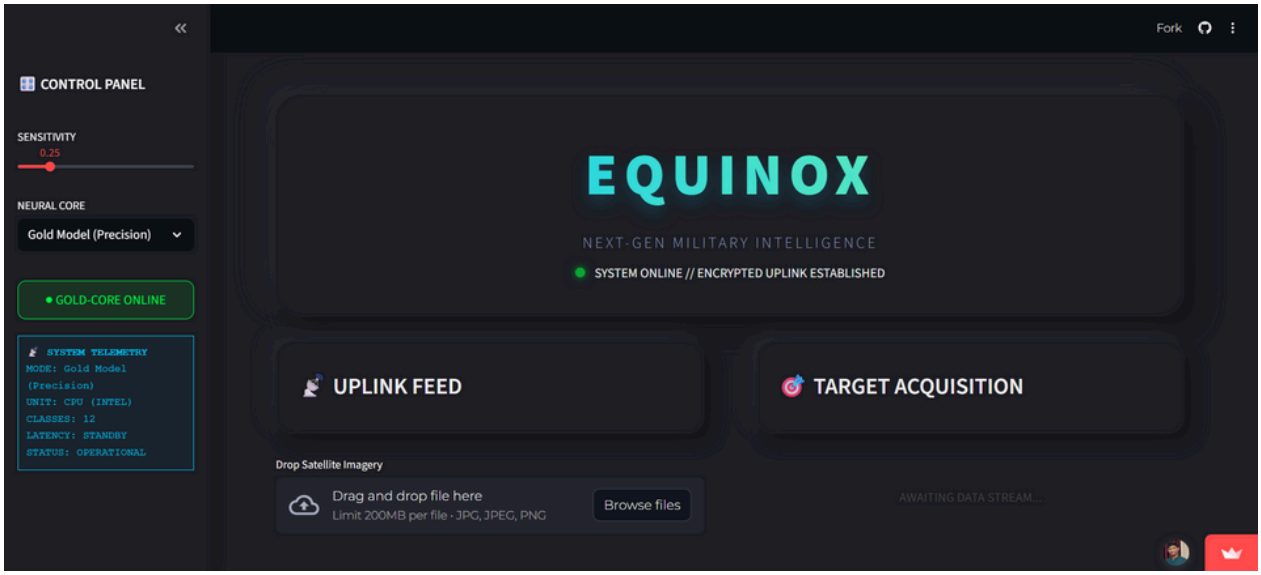
CONCLUSION

- *Class imbalance*: Solved via synthetic data engineering
- *Accuracy vs speed:* Solved via Dual-Core ensemble inference
- *Deployment*: Solved via OpenVINO optimization & containerization using Docker

---

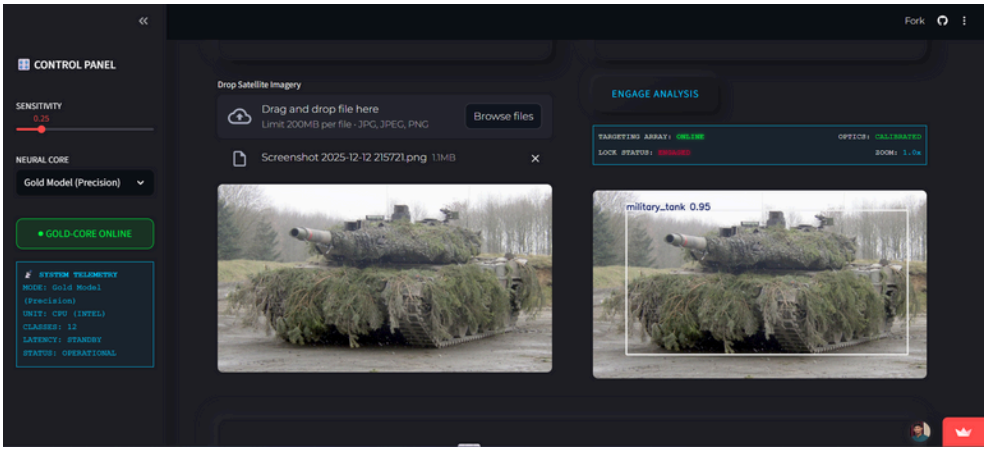
EQUINOX - USER INTERFACE

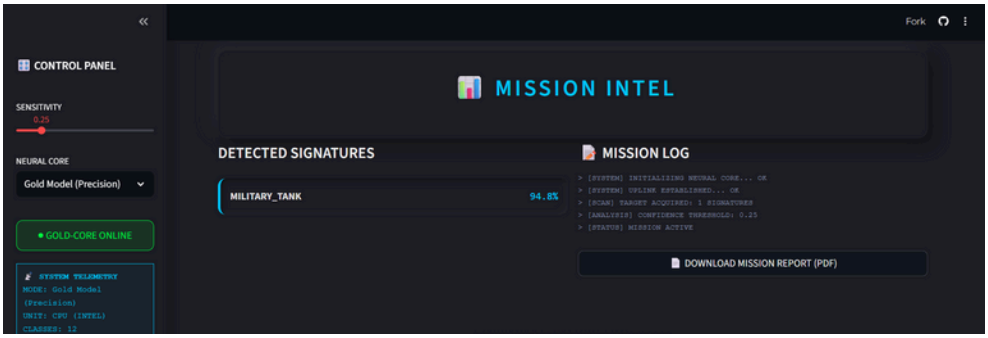
WEBSITE LINK : http://equinox25.streamlit.app/



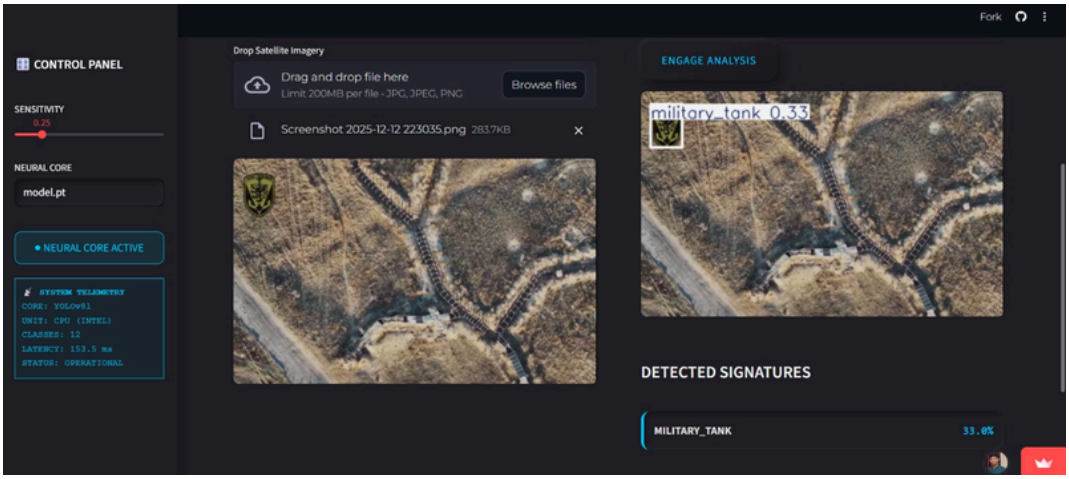
TEST CASES AND RESULTS

TEST CASE 1- MILITARY TANK✅





TEST CASE 2 : TRENCH



WHY DID IT GO WRONG : The model briefly misidentified the top-left logo as a military tank with a low 33% confidence. This happened because the emblem's sharp, high-contrast edges resemble tank-like features, confusing the CNN. Since the model was trained only on clean satellite imagery, it tries to force any unfamiliar object into its known classes—an expected out-of-distribution behaviour. The low score shows it was unsure, and increasing the sensitivity threshold to 0.40 instantly removes this false positive. This edge case actually validates our design: the threshold slider effectively filters weak detections, and it highlights the importance of cropping overlays/watermarks before analysis.