edunet foundation

# Smart Traffic Detection via YOLOv7

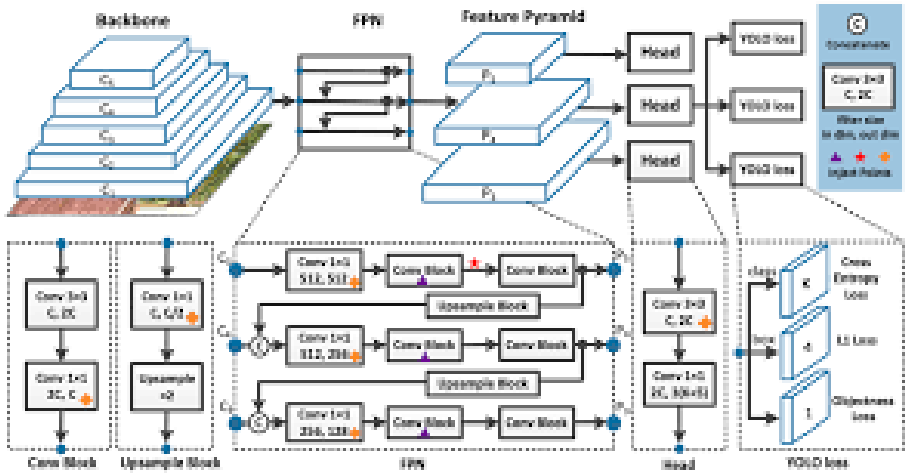
## Learning Objectives:

- To aid for **sustainable development** using **Machine-Learning**, especially, **Deep-Learning** models, to classify objects/actions.
- To gain hands-on experience in creating a **multi-process** pipeline that helps in sustainable **city-development.**
- Analyze how automation and data-overlays contribute to **SDG 11** (**Sustainable Cities and Communities).**
- Contribute to the analysis of **Fuel-Consumption** and **Emission** at traffic junctions, thereby helping in implementation of techniques that reduces them.


**Source :** www.freepik.com/

## Tools and Technology used:

- **OpenCV**(Initializes camera for real-time detection).
- **YOLOv7**(Trained with custom-made dataset).
- **Python**(Coding and Libraries utilized for architecture).
- **Tensorboard**(To track training rate and log similar criteria).
- **LabelIMG**(For creating custom dataset).
- **Tkinter**(GUI Develeopment).



## Methodology:

- **Custom Dataset Preparation –** Annotated vehicle classes (Truck, Bus, Car, Motorcycle) using **LabelImg** via hand-picked traffic footage.
- **Model Training & Optimization –** Trained YOLOv7 on the custom dataset using **PyTorch,** monitored via **TensorBoard**, and exported weights.
- **Real-Time Detection-** Deployed **detect.py** to process live or recorded feeds and save class-wise label with confidence scores.
- **Threshold-Based Filtering –** Implemented a filter module **(class_filter.py)** to compute rolling averages, and apply per-class thresholds and write structure JSON thresholds.
- **Signal Decision & Overlay –** Designed a decision module **(dig_tr_sgn.py)** to read JSON counts, and to trigger signals based on thresholds.
- **GUI-Orchestrated System Integration -** Built a Tkinter-based control panel to launch, monitor, and terminate subprocesses, enabling modular, user-friendly orchestration of the full pipeline.

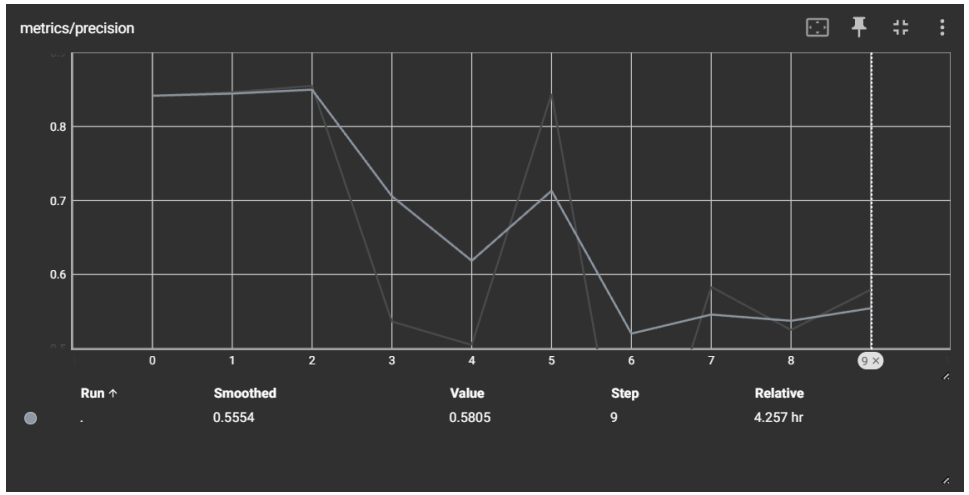
## Problem Statement:

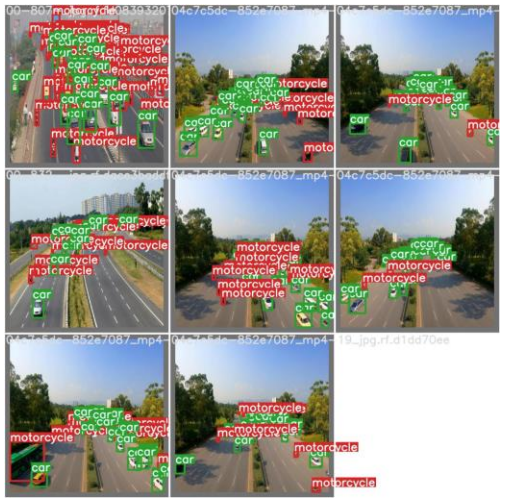
- **Urban intersections** suffer from inefficient traffic signal timing, leading to prolonged idle times and increased fuel consumption.
- **Manual traffic control** systems lack real-time responsiveness, making it difficult to adapt to fluctuating vehicle densities and class types.
- **High congestion levels** contribute to elevated carbon emissions, noise pollution, and commuter frustration, undermining sustainable mobility goals.
- **Existing surveillance** infrastructure often lacks intelligent automation, resulting in underutilized data and missed opportunities for optimization.
- There is **limited integration** between detection, decision-making, and visualization, making it hard for operators to monitor and act dynamically.

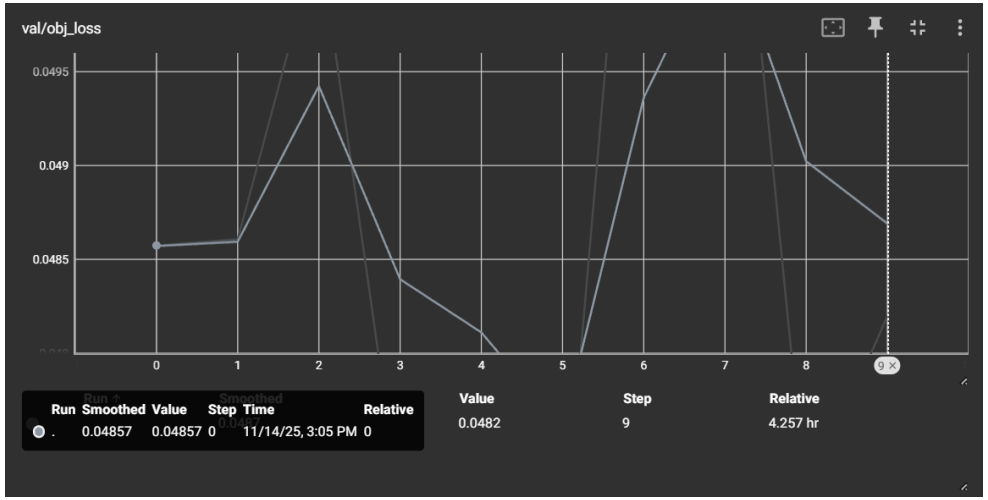
## Solution:

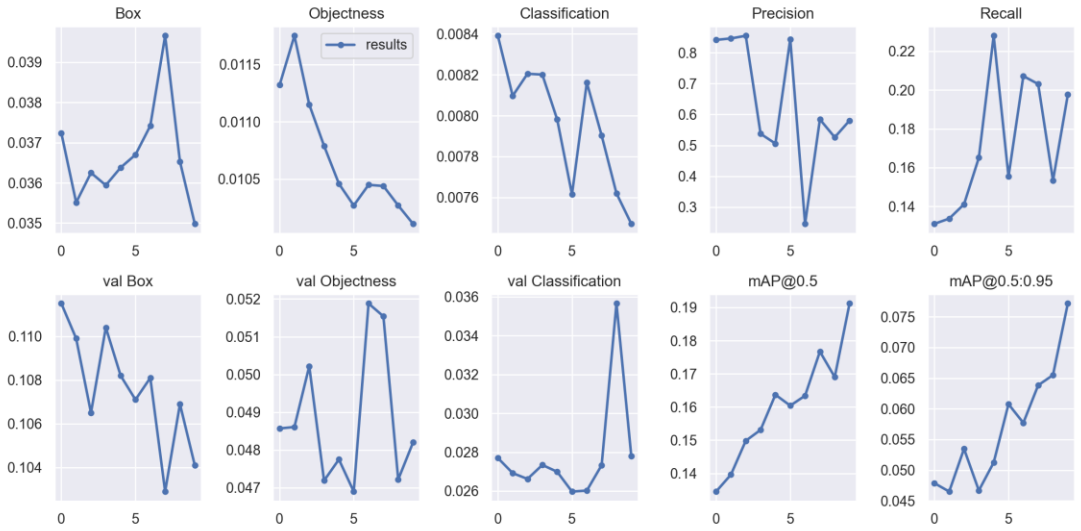
- **Real-Time Vehicle Detection with YOLOv7** - Deployed a custom-trained YOLOv7 model to identify and classify vehicles (truck, bus, car, motorcycle) from live or recorded video feeds with high accuracy.
- **Rolling Average-Based Threshold Filtering -** Implemented a dynamic filtering mechanism that computes class-wise rolling averages over recent frames to smooth out noise and ensure reliable signal decisions.
- **Automated Traffic Signal Decision Logic -** Designed a rule-based system that triggers GREEN signals when specific vehicle class thresholds are met, reducing idle time and improving flow efficiency.
- **Live Overlay with Visual Feedback -** Used OpenCV to render real-time overlays on the video feed, displaying signal status, vehicle counts, and trigger reasons for operator transparency.
- **Modular, Multi-Process Architecture -** Separated detection, logging, and overlay into independent scripts communicating via JSON, enabling parallel execution and easy scalability.
- **User-Friendly GUI for System Control -** Built a Tkinter-based dashboard to launch, monitor, and stop the pipeline with configurable inputs, making the system accessible and operator-ready.

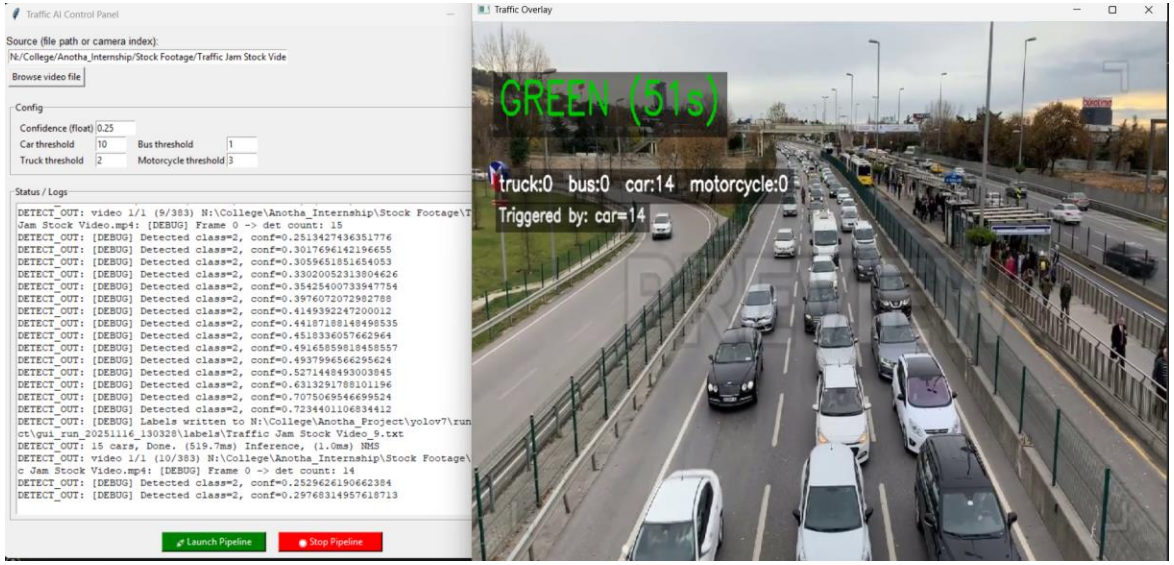
# Screenshot of Output:





## Conclusion:

- **Successfully built a real-time traffic detection** and control system and integrated object detection, threshold logic, and visual overlays into a modular, fully functional pipeline.
- **Demonstrated the power of AI in urban mobility optimization** and displayed how smart automation can reduce congestion/traffic.
- **Engineered a scalable, multi-process architecture,** thereby enabling parallel execution of detection, logging, and visualization, adaptable to multi-feed deployments.
- **Aligned technical innovation with sustainability goals** resulting in reduced idle emissions and fuel waste, contributing to cleaner, smarter cities in line with **SDG 11.**