**Real-Time Video Analytics: Detection of Missing and Newly Placed Objects**

**Objective**

The objective of this project is to build a real-time video analytics pipeline capable of identifying and tracking objects in a scene using computer vision. It further aims to detect two critical events:

* **Missing objects**: Objects that were once present but have been removed.
* **Newly placed objects**: Objects that appear in the scene after a certain point in time.

This functionality can be useful in surveillance, inventory monitoring, and anomaly detection.

**Tools & Technologies Used**

* **YOLOv5** (Pretrained model from Ultralytics via torch.hub)
* **Norfair** (Lightweight object tracking library)
* **OpenCV** (Video stream handling and visualization)
* **PyTorch** (Backend for YOLOv5)
* **Python 3.x**

**System Architecture**

1. **Object Detection**:
   * YOLOv5 is used to detect objects frame-by-frame.
   * Classes are filtered based on relevance (e.g., ignoring ‘person’ during removal detection).
2. **Object Tracking**:
   * Norfair tracks detected objects using Intersection over Union (IoU) as the distance function.
   * Each tracked object is assigned a unique ID and tracked across frames.
3. **Disappearance and Movement Logic**:
   * When an object ID is not detected for a predefined number of frames, it is considered ‘removed’.
   * Movement is tracked using the centroid of bounding boxes. If an object is stationary and disappears, it is flagged.
4. **Visualization**:
   * Bounding boxes are drawn for detected and tracked objects.
   * Removed objects are shown with red boxes and a ‘REMOVED’ label.
   * FPS is calculated and displayed on-screen for performance monitoring.

**Key Features**

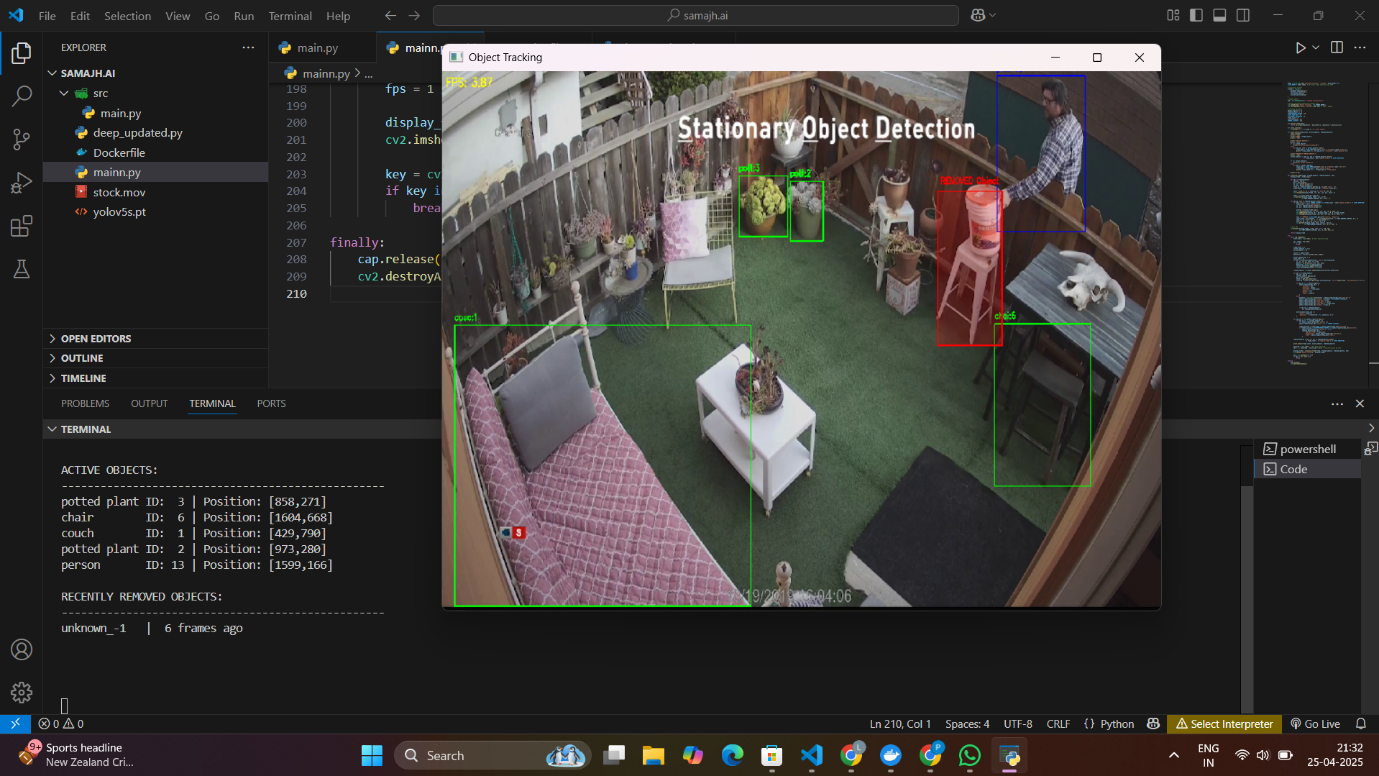
* **Real-time FPS Optimization**:
  + FPS is calculated using frame processing timestamps to assess and improve pipeline speed.
  + The model used (YOLOv5s) is lightweight for faster inference.
* **Dynamic Object Monitoring**:
  + Tracks movement and persistence of objects using bounding box centers.
  + Handles visual alerts for removal with a configurable alert duration.
* **Console Summary**:
  + A clear tabular printout in the terminal displays:
    - Current frame
    - List of active objects
    - Recently removed objects with timestamps

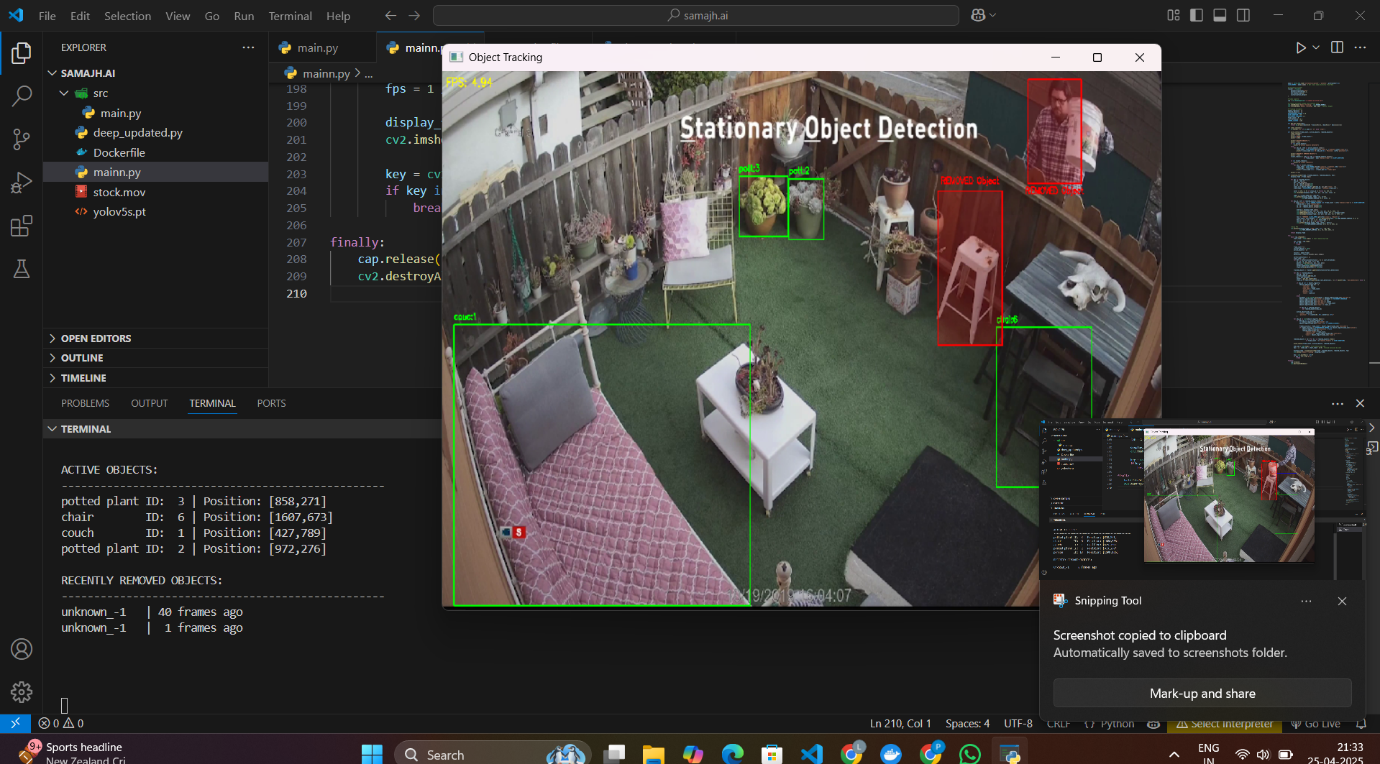
**Challenges & Optimizations**

* **Challenge**: Detecting object removal accurately without misclassifying temporary occlusions.
  + **Solution**: Set a disappearance threshold and only flag removal after a defined number of frames.
* **Challenge**: Maintaining high FPS with detection + tracking in real-time.
  + **Solution**: Use YOLOv5s and optimize frame handling and tracking update intervals.

**Future Improvements**

* Integrate **GPU acceleration** for better inference speed (currently tested on CPU).
* Add **new object placement detection** logic.
* Use **motion tracking filters** (e.g., Kalman Filter) for smoother tracking.
* Export removal alerts to a **log file** or cloud dashboard.





Demo video link : [demo video link](https://drive.google.com/file/d/1hBux8__ufPYx9a-ICcOvhVeq945n_UQy/view?usp=sharing)