1. Model Used and Why?

—> The chosen model for vehicle detection in this project is **YOLOv3-Tiny**, a lightweight and faster variant of the YOLO (You Only Look Once) family. YOLOv3-Tiny is well-suited for real-time applications and embedded systems due to its reduced complexity and faster inference speed while maintaining reasonable accuracy. It processes frames efficiently, making it ideal for video-based vehicle detection, even on hardware with limited computational resources.
The model is pre-trained on the **COCO dataset**, which includes object classes such as cars, buses, trucks, and motorcycles. This allowed the system to identify common vehicle types without requiring custom training.

2. Pipeline / Process

—>

Model and Class Loading:

- `YOLOv3-Tiny` weights and configuration (`.cfg` and `.weights`) files are loaded using OpenCV's DNN module.
- `YOLOv3-Tiny has 13 convolution layers.`
- COCO class names are loaded from `coco.names`.

Video Processing Loop:

- Each frame from the input video (`855981-hd_1920_1080_30fps.mp4`) is read and preprocessed into a blob.
- The blob is fed into the YOLO network for forward propagation to detect objects.

Bounding Box Extraction:

- Predictions are parsed to extract bounding boxes, class IDs, and confidence scores.
- A confidence threshold of 0.88 is applied to filter weak detections.
- Non-Maximum Suppression (NMS) is used to eliminate redundant overlapping boxes.

Custom Vehicle Classification:

- Detected vehicles are further classified based on bounding box size heuristics into categories such as:
  - o *Ambulance/Small Bus / Big Bus*
  - o *Auto (Three-Wheeler)*
  - o *2-Axle / 3-Axle / Multi-Axle Trucks*
- This provides more granular insights beyond default COCO classes.

Display and Counting:

- Bounding boxes and labels are drawn on each frame.
- Vehicle counts per category are updated in real time and displayed.
- FPS is calculated and shown to monitor performance.

3. Challenges and Assumptions

Challenges:

- **Size-Based Classification Heuristics**:
  The additional categorization (e.g., "3-Axle Truck") relies on bounding box area, which may not scale proportionally due to camera angle, perspective distortion, and vehicle orientation.

- **False Positives**:
  As confidence score is higher so it may leave some vehicles while counting and classification.

Assumptions:

- The video is taken from a fixed angle with consistent lighting and a visible stream of traffic.

- Only a fixed set of vehicle types (e.g., car, bus, truck, ambulance) is of interest.

4. Accuracy Notes and Limitations

- **Confidence Thresholding**:
  A high confidence threshold (0.88) reduces false positives but may miss partially occluded or small vehicles.

- **YOLOv3-Tiny Limitations**:

  - While fast, it sacrifices some accuracy compared to the full YOLOv3.

  - It may underperform in complex scenes with dense traffic or occlusions.

- **No Model Retraining**:
  The system depends entirely on the pre-trained COCO classes. For better performance, fine-tuning on a traffic-specific dataset could improve results.