Vehicle Detection and Tracking with Lightweight Object Detection Model Based on YOLOv11n

Student Name: ChingHsuan Tsai

Student ID: M01001570

Supervisor: Francois Chadebecq

Outline

- Background
- Arguments
- Aim and Objectives
- Approach
- Results
- Conclusion and Evaluation
- Future Work

Background

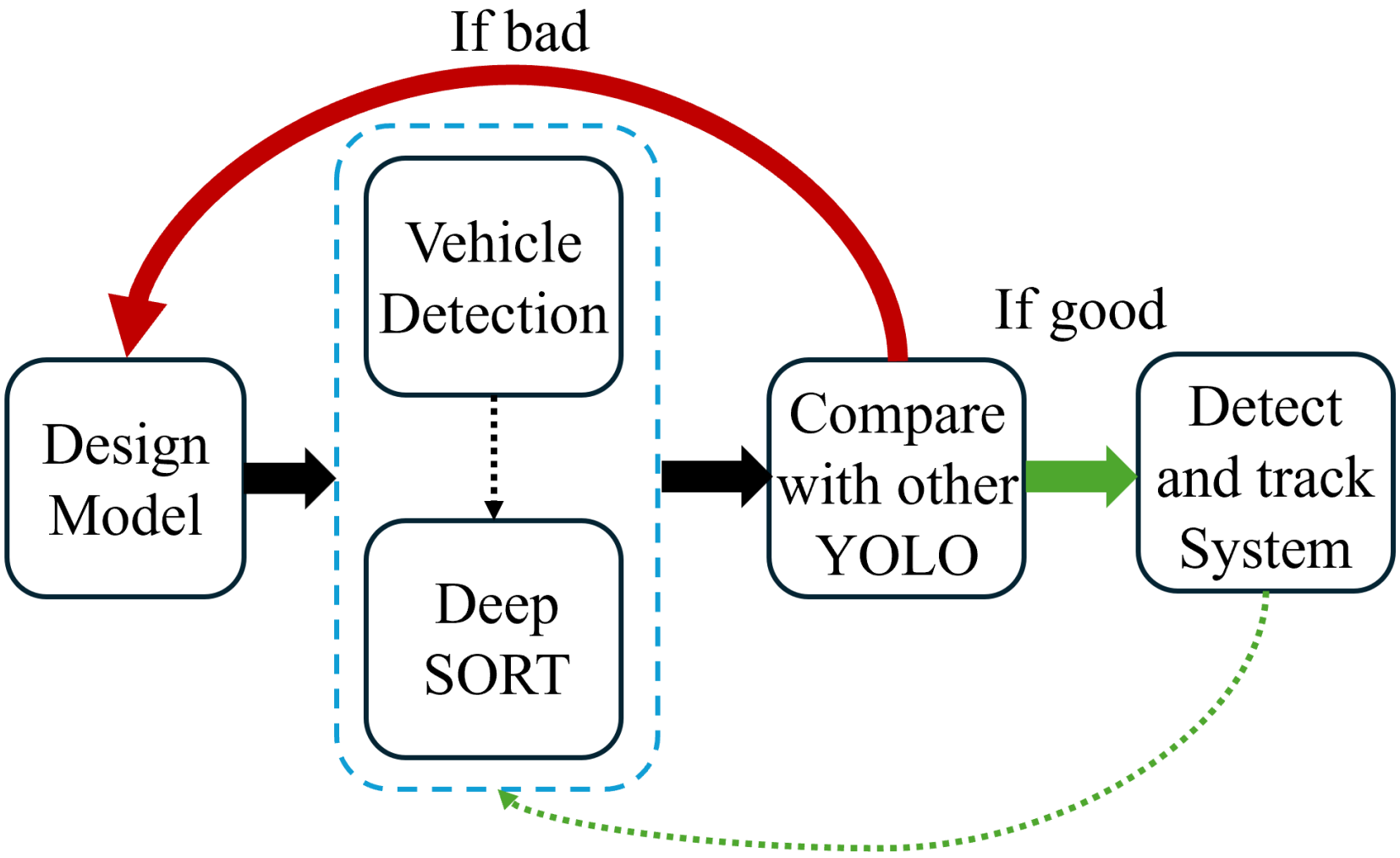
1. Increasing vehicle usage → traffic congestion and accidents.
2. Real-time vehicle tracking helps optimize traffic flow and safety.
3. Deep learning-based object detection + tracking = mainstream trend.

Arguments

1. **Fast-moving vehicles** are hard to track accurately with heavy models.
2. **Network instability** in developing areas limits cloud-based solutions.
3. **Dual-model systems** (object detection + tracking) are too resource-intensive for embedded platforms.

Aim and Objectives

To develop a lightweight vehicle detection and tracking system for resource-constrained environments, the picture is my Objectives:



Approach

Tools & Environment (1)

This workstation is the hardware specifications used in the experiment.

Hardware specifications

Component \ Platform	**Workstation**
GPU	NVIDIA GeForce GTX 4070 Ti
CPU	Intel (R) Core(TM) i7-13700 CPU @ 3.60GHz
Memory	64GB
Storage	1TB× 1(HDD)

Tools & Environment (2)

◆ Software used in the experiment and their versions.

Recipe of Packages

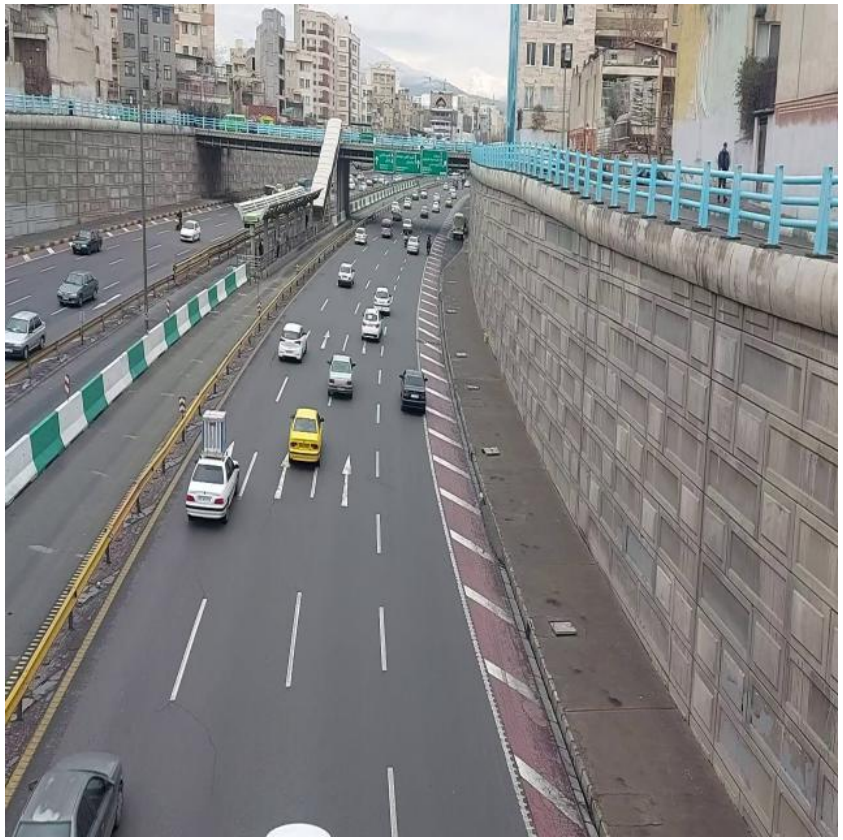
Software	Version
Python	3.9.21
Pytorch	1.9.1
Anaconda	23.7.2
Ultralytics	8.3.76
Deep_sort_realtime	1.3.2

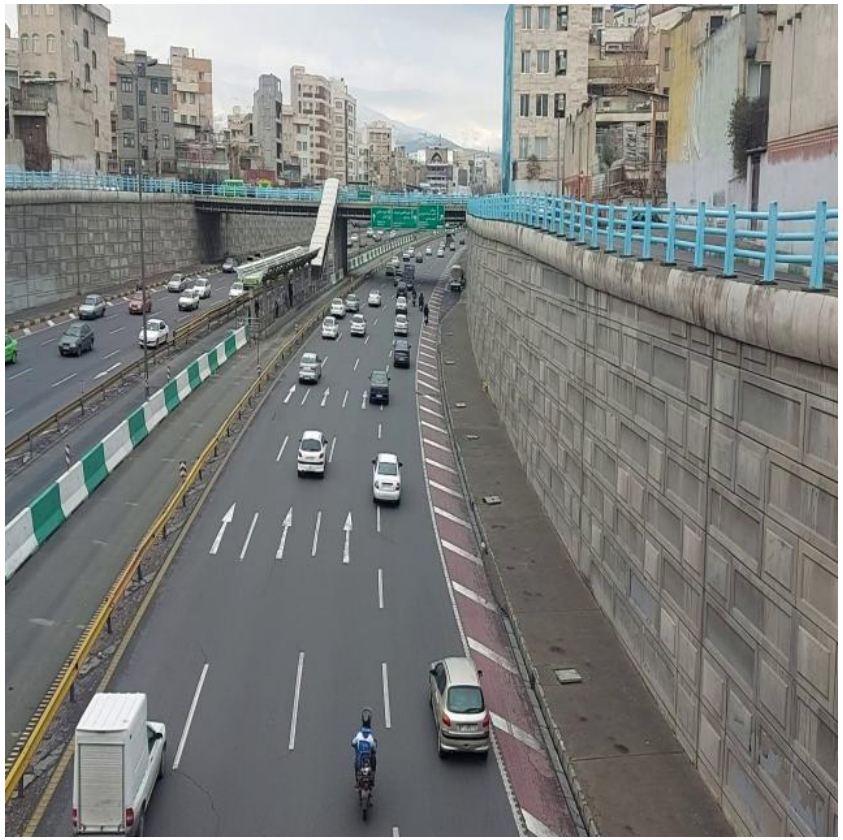
Dataset

Image size: 640×640

Train/Val/Test = 70/20/10

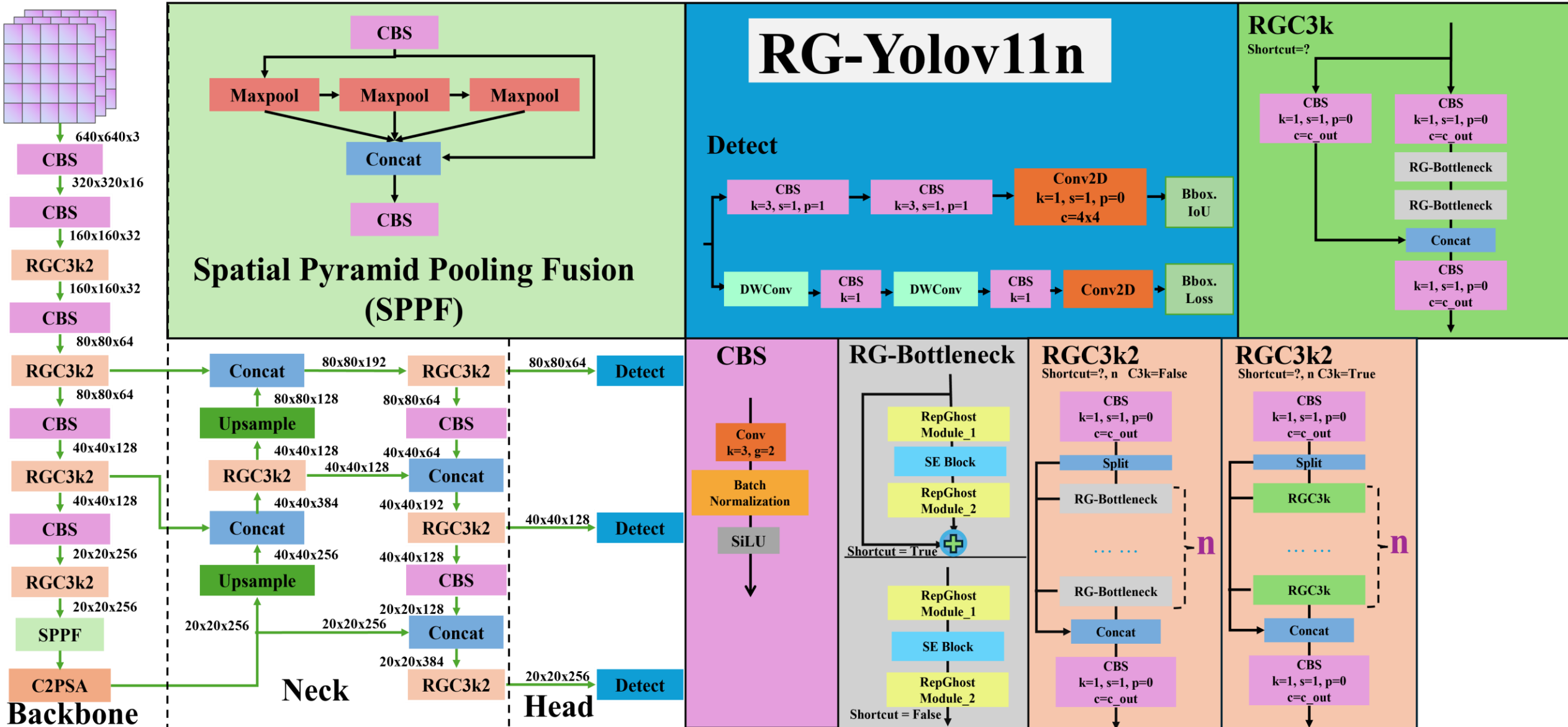
Source: https://www.kaggle.com/datasets/pkdarabi/vehicle-detection-image-dataset?resource=download





Model Architecture

Replace All the Bottleneck to RG-Bottleneck



Results

Performance

Evaluate / Model	Param	FLOPs	Precision/%	Recall/%	mAP@50/%	FPS
YOLOv5n	2,503,919	7.1G	71.3	68.9	74.6	24.26
YOLOv8n	3,006,623	8.1G	83.9	73.1	75.7	30.86
RG-YOLOv8n	2,230,335	6.1G	75.8	67.2	69.3	40.13
YOLOv11n	2,538,127	6.3G	86.0	73.1	78.8	34.15
RG-YOLOv11n (our)	2,276,735	6.2G	83.7	68.9	74.5	37.26

Demo Video

Conclusion and Evaluation

1. Successfully implemented vehicle detection and tracking system.
2. Successfully implemented a lightweight vehicle detection model.
3. Our model improves FPS and reduces parameters significantly.
4. Precision is slightly reduced but still acceptable for real-world applications.

Future Work

1. Introduce advanced attention mechanisms (e.g., CBAM, ECA) to improve accuracy.
2. Explore alternatives to Deep SORT (e.g., a lighter version of StrongSORT).
3. Consider hardware deployment (e.g., Jetson Nano) for practical validation.

Thank You