Traffic Congestion Reduction through Real-time Object Detection: Analyzing the Effectiveness of different CNN models such as Mask RCNN, SSDNet and Yolo.

MSc Research Project
MSc in Artificial Intelligence for Business

Sanjay Rastogi
Student ID: x23160977

School of Computing
National College of Ireland

Supervisor: Anderson Simiscuka

National College of Ireland
Project Submission Sheet
School of Computing

Student Name:	Sanjay Rastogi
Student ID:	x23160977
Programme:	MSc in Artificial Intelligence for Business
Year:	2024
Module:	MSc Research Project
Supervisor:	Anderson Simiscuka
Submission Due Date:	12/08/2024
Project Title:	Traffic Congestion Reduction through Real-time Object Detection: Analyzing the Effectiveness of different CNN models such as Mask RCNN, SSDNet and Yolo.
Word Count:	7570
Page Count:	25

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature:	
Date:	15th September 2024

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST:

Attach a completed copy of this sheet to each project (including multiple copies).	□
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	□
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	□

Assignments that are submitted to the Programme Coordinator office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Traffic Congestion Reduction through Real-time Object Detection: Analyzing the Effectiveness of different CNN models such as Mask RCNN, SSDNet and Yolo.

Sanjay Rastogi
x23160977

Abstract

One of the major usage of this technology is in Vehicle detection and counting. Vehicle counting is the process of counting the number of different types of vehicles that have crossed a particular area/line or entered into a particular zone. The former is an extension of the latter, and there is no nice way of putting this, but these systems are practical as hell. For instance, the implementation of other classifiers to distinguish between diverse sorts of products like trucks, automobiles, and bicycles is an example. Other areas include traffic management and observation and-direction, highway and safety surveillance and directing, Urban planning and mapping, tracking of specific vehicles, automatic number plate recognition, real-time traffic information, tolls, congestion and related data, crowd/ pedestrian counting, facial recognition and alignment, and analysis to mention but a few. In our attempt to do this research, we are aim for a detailed evaluation of several objection models in traffic signals that are effective in counting the number of vehicles. Besides the accuracy of the vehicle calculation the priority is also on computation and space of the algorithms so that on can fit into the real time analysis. Performing a through analysis we realized that YOLOv10 together with YOLOv8 had a better performance compared to the rest of the models. Since this whole pipeline will act as a response system in integrating with the devices present in the traffic, comparing the overall performance in terms of the number of objects detected for YOLO against RetinaNet, SSDNet, and MaskRCNN, it was evident that YOLO fared better not only in detection but also the time taken to conduct the detection (higher FPS).

Keywords: Object Detection, Frame per Second, YOLO models, Real time analysis, Cloud Framework, Automated traffic management system

1 Introduction

The worldwide market for intelligent traffic management and vehicle detection systems was estimated at USD 10,423.7 million in 2022 and is projected to grow at a compound annual growth rate (CAGR) of 13.8% until 2030 (Iftikhar et al., 2023). The primary factors contributing to this rapid increase may be linked to the increasing demand for traffic-related information from drivers, as well as government programmes aimed at improving traffic, accident, and infrastructure management. In this blog, we will explore the techniques, executions, and algorithms used for vehicle detection ((Ng and Kwok, 2020).

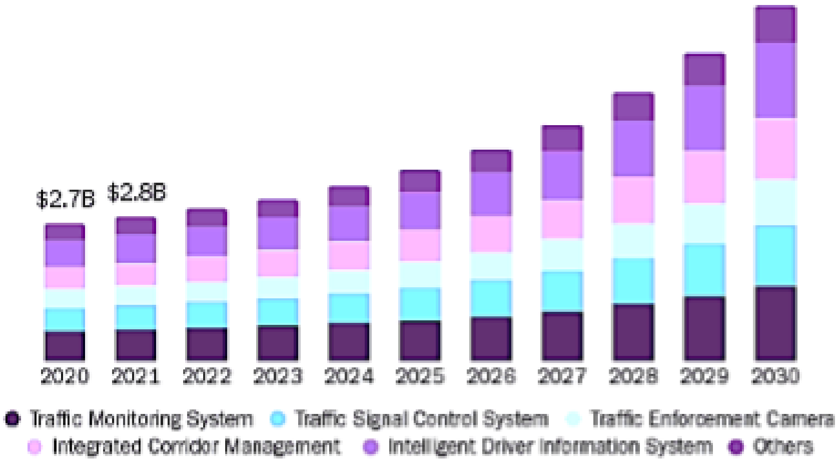


Figure 1: Projection of the Vehicle detection system market size from 2020 to 2023 (Source: GrandViewResearch)

In order to lessen the negative effects on the environment, increase safety, improve reaction times in emergencies, and decrease traffic congestion, smart traffic management systems are essential in today's cities. Travel times, fuel consumption, and CO_2 emissions are all reduced by these systems' traffic flow optimisation, which is made possible by cutting-edge tech like AI and Internet of Things sensors. This minimises the loss in economy by lowering the accidents numbers and road wear and tear due to humongous traffic, and this as well give real-time data for improved city planning and infrastructure maintenance. By enhancing the efficiency, safety, and environmental friendliness of urban transportation, these systems ultimately enhance the quality of life (Wang et al., 2023).

1.1 Aim

Using advanced technologies like computer vision aided deep learning models and machine learning, artificial intelligence (AI), and Internet of Things (IoT) sensors (SenthilPandi et al., 2023), the project aims to develop and implement a comprehensive response system for smart traffic management system. This system will help in optimise traffic flow, enhance safety, reduce environmental impact, and improve overall urban mobility. City planners will be able to make informed decisions about infrastructure construction and maintenance with the use of real-time data and predictive insights provided by this system (Barbosa et al., 2023). It also ensures speedy emergency response and saves money by reducing congestion and accident rates. In the end, we want to build a system of urban transport that is safe, efficient, and environmentally friendly so that everyone living in the city may enjoy a better quality of life.

1.2 Motivation

One consequence of our modern era's rapidly expanding human population is the accompanying surge in the number of individuals driving their own personal vehicles. With traffic congestion getting worse by the day, finding a practical solution has become a top priority. Currently, there is a lack of appropriate solutions that might help regulate and alleviate traffic-related difficulties. The limitations that are now in place make it harder for us to detect incidents and start acting fast in the event of an accident, which in turn makes it take longer to fix the problem (Ramana et al., 2023). Because it took so long to figure out what happened and where the problem originated, the situation has gotten much worse. This technology should concurrently monitor for the manipulation of the traffic signals and delivers quick notifications to authorised workers in the case of an occurrence (Madhavi et al., 2023). A faster and more effective response is possible because of this. Consequently, it is feasible to respond concurrently with the incident, so mitigating its effects.

1.3 Research Objective

To do extensive research on the existing newer and older deep learning algorithms to find which all algorithms are efficient, faster and lesser storage required to run. This will help us in not only finding a place where a detailed analysis for all the algorithms is used but also to check how these algorithms are efficient in the space as well as the computational time. We would also try to do this analysis on three different kinds of standard dataset like COCO, KITTE and TT100K and then apply on a real time dataset. This will help us in making a generic object detection algorithm to be used as a response system of the traffic management. Now this response system then can help in providing additive insights for an automated traffic management system which is our primary objective and for other problems like home surveillance, parking monitoring etc. In case of the traffic management systems, we can put conditions for the green time to be shown for the effective traffic movement.

1.4 Research Question

RQ: How to effectively detect the objects with minimal computation time(FPS: Frame per second) and less space, that can be deployed either in Cloud or Edge?
Solution: We will analyse different models like RCNN, Mask RCNN, YOLOv3-YOLOv8, SSDNet and RetinaNet to finalise the model which is able to detect objects and in minimum time so that the real time analysis is not hampered. Since the videos will be processed and send to the cloud or the model which trained and saved on the edge device which is the traffic signal system to detect in a fraction of second. We propose to use our solution which is an ensemble of video processing models capped with a forecasting model to take a decision. The vehicles detected per timeframe will be the input to the forecasting model. This will take care of many boundary conditions like storm, congestion (with Q-Learning), blur visions etc. We will also implement the detection of special vehicles like ambulances so that immediate action can be taken for.

2 Literature Review

2.1 Recent advancement in object detection

Real-time object identification is addressed in the study by (Zhao et al., 2024) and they propose the limitations of the real-time object identification models, and particularly the models derived from YOLO and DETR families. In some cases, YOLO models maintain an ideal scale of speed and accuracy; however, the NMS operation acts as a disadvantage. Although DETR models eliminate NMS operations, they are highly time-consuming to apply in real-time applications. In this paper, applying the RT-DETR-R101 model, on a T4 GPU (Zhao et al., 2024), the mAP reached 54. On this scale, the MViT-3 achieved 3% AP at 74 fps, while the RT-DETR-R50 was recorded to perform at 53%. 1% AP at 108 fps, which is another example of excellent results for both graphics cards. I got it 2 nearing the end of a procession in which one candidate beat the other by a margin of 2. Moreover, this model also beats DINO-R50 in terms of the performance score, which achieved 2% AP with a factor of 21 out of the total 100.

This paper (Pu et al., 2024) is concerned with the problem of ranking precision in DETRs. The paper mainly focuses on the separation between the classification deficit score and localization precision so as to have a proper way of identifying high quality items. The authors introduce Rank-DETR, which is a set of rank-aware designs' goal is to enhance the DETR-based object detectors, especially in cases of high IoU thresholds. Rank-DETR incorporates both of these major improvements. On average the model achieved 5% more in AP than the H-DETR model. Likewise, when applying Swin-T, it provided an uplift of 2. 7% improvement in AP75. Here, it is crucial that the experiments of the study embrace numerous circumstances, stressing the effectiveness of the method in generating more elevated AP scores where IoU is high, which offers a powerful solution to the challenges of modern object detection.

The new object detection framework was called DiffusionDet as described in this (Chen et al., 2023) paper. This method reduces the object recognition problem to denoising diffusion. Thus, the problem of enhancing the stability and flexibility of the object detection model without spending a lot of time on retraining remains the most acute. That DiffusionDet's goal is to view object detection as generative is the key that underlies its methodological approach. This is achieved by a process of continually improving on the current design of the product. Some of the badges of the system include the repetitive evaluation and the box management. They allowed the system to adapt rather than retune, several properties of these systems are as follows: On the COCO dataset, CrowdHuman, and LVIS, the proposed detectors performs better than the Faster R-CNN and DETR detectors. DiffusionDet can reach 46. Specifically, on 8 AP with a ResNet-50 backbone, the performance surpasses previous object detection benchmarks. The ability to vary assessment boxes and the number of iterations with the dataset properties accounts for the model's high performance.

OWOD is the problem of detecting objects that were not seen during the training of the model, which previously are labeled as known objects or ignored as background by traditional methods; Probabilistic Objectness Open World Detection Transformer (PROB) Zohar et al. (2023) tackles this issue. To address this, PROB proposes a new solution that discerns a probabilistic objectness head with the deformable DETR model. Notable new components include a Probabilistic Objectness Head for estimating the probabilities of objectness without assuming that the mismatched object suggestions are the background.

2.2 Recent development in traffic object detection

The paper (Zhang et al., 2023) tackles the issue of recognizing traffic signs at different scales in the accurate and real-time manner based on the object detectors, mostly in the case of small objects. In order to counter these issues, the authors endeavour to present the CR-YOLOv8 model, which is an improve version of YOLOv8. The changes are the Convolutional Block Attention Module for better spatial and channel features in the feature extraction phase and the Receptive Field Block for better feature diversity in the feature fusion phase that does not disturb the computational budget much. Furthermore, by using multi-scale loss, the model loss function is trained to reduce multi-scale object detection in training. Such technical specifics are the utilization of the TT100k dataset, which comprises 100,000 images and thirty thousand traffic sign instances under different circumstances. Training is done with a batch size of 32 for 200 epochs and when ImageNet weights are used for initialization, carefully chosen hyperparameters are used. The experimental setup adopted is on Linux systems with NVIDIA Tesla A100 GPUs and implemented with PyTorch. This proves that CR-YOLOv8 has reduced the average detection accuracy by 2% as seen from the performance results above. Therefore, the accuracy of recognizing small objects was increased by 1% while the recognition took only 3% of the time. 6% better than the baseline of YOLOv8s.

The rationale of the presented study (Li, Sun, Zhang, Feng, Wu and Li, 2023) is to enhance the YOLOv5s algorithm to estimate minor traffic things in perplexing road conditions. The proposed SPD-Conv CNN Module increases the chances of identifying low-resolution pictures and relatively smaller objects by utilizing space-to-depth and non pooling layers. NAM can, therefore, help to focus on the significant aspects, which in turn results in accurate detection. Therefore, experimental results analysed to identify key differences reveal a mean of 7. The proposed method enhances the detection accuracy by 1% compared to the YOLOv5s as demonstrated when tested on a more complex traffic scene's dataset. This is way superior to any other technique of object detection that is commonly employed. The proposed work improves the number of parameters and calculations compared to YOLOv5x model though the level of object detection accuracy is satisfactory. This shows it is well trained and capable of negotiating tricky traffic conditions in short timespans. General traffic items and as well as several difficult samples were incorporated in the sets to guarantee a comprehensive and credible assessment.

The challenge involved in the identification of small objects especially cars from surveillance videos is a major issue. This research (Akhtar et al., 2022) solves the problem under consideration by proposing DenseNet-201 as the principal network for the YOLOv2 algorithm. It was this enhancement that sought to enhance feature extraction by becoming more efficient while at the same time cutting on the parameters. The advanced YOLOv2 technique was trained with the Kaggle and KITTI vehicle datasets and cross-checked with Pascal VOC and MS COCO datasets. The modified model provide the intersection over union (IoU) of 97 percent. 06% while the network has shown an identity recall of 90%, and an average precision (mAP) of 97. 51% which is higher than the previous approaches of vehicle detection. The outcomes were continuously accurate, while the categories such as the Taxi and the Van reached mAPs of up to 0.98. 9%. The assessment unveiled that the incorporated method provided enhanced results than popular algorithms like Faster RCNN, SSD, and various versions of YOLO. The type of cross-validation we used provided evidence of the model's stability with Pascal VOC yielding mAP of 81% and COCO of 75. 1%.

This article (Lai et al., 2023) tries to address the challenge that arises due to the change in the weather condition, topological features limiting vision and differences in illumination of roads so as to explore the possibility of the recognition of traffic signs. To overcome this problems, the researchers created an improved TT100K dataset commonly referred to as the TT100K-Enhanced dataset. To introduce difficult cases, the data was augmented by fog, noise, snow, occlusion, and blur. Due to the small size and low contrast of objects in some conditions, the new network called STC-YOLO was introduced to improve the existing YOLOv5 network. In the evaluation of the advanced TT100K set, STC-YOLO outperforms the demonstrated results and the most recent state-of-the-art practices in both TT100K and CCTSDB2021 datasets. STC-YOLO achieved a 9. An approximately 3 percent better increase in the mAP scores than YOLOv5 is achieved by the proposed modifications. In ablation studies, the efficacy of each module was proved with improved detection accuracy and it's resilience in testing conditions. This work of adding a side output pathway to the existing structure increases the accuracy of detecting small objects and achieves real-time detection at the same time. Hence, it can be considered as a dependable option particularly for cases that require self-driving.

ITS and self-driving cars hinge on their ability to recognize timelike traffic signs in complex conditions. In the current work,(Li, Wang and Wang, 2023) assumes the task of elaborating on these indications. Due to the necessity to enhance the identification ability of small objects, the authors suggest adopting an improved model of YOLOv7, known as SANO-YOLOv7. In order to strengthen the collection of feature information, in this paper, we embed a module called self-attention and convolutional mix (ACmix). Performance evaluation strategies such as recall, precision, and mean average precision (mAP) were used in measuring the performance. SANO-YOLOv7 achieved a 88 % mAP, SN for real-time detection at 33 fps in the test video. 7 for the IoU score of over 0. 5, these are five. 3% improvement than the reference YOLOv7 model. Comparing with other models like YOLOv3, YOLOv5, YOLOv6, SSD, Faster-RCNN, SANO-YOLOv7 demonstrated significant improvements in the detection precision and inference time. The developed technique is relatively favourable for examples like finding traffic signs in complex environments because of features like its high learning ability, fast convergence and high mAP scores. The post-processes include exploring more compact model derivatives for the mobile application adaptation, increasing the model's ability to generalize, and expanding the database.

2.3 Research Niche

As we investigate the above papers, we find a detailed research and analysis on different object detection models on the video dataset is not available. This creates a space in which we can have a look on to these models and check the performance not only in terms of the mAP or detection of a particular objects but also into the computational speed and the space that it takes. This will help in the terms of how this solution is used whether servers which has enormous computational power and space, but the cost of utilisation makes it expensive. The idea behind this research is to have a solution which can handle all the different computational problems including the "solution blackout"

Name	Authore	Model	Dataset	Result
DETRs Beat YOLOs on Real-time Object Detection	Zhao, Y., Lv, W., Xu, S., Wei, J., Wang, G., Dang, Q., Liu, Y. and Chen, J.	RT DETR-R101, RT-DETR-R50	COCO	54.3% AP at 74 FPS ; 53.1% AP at 108 FPS
Rank-DETR for High Quality Object Detection	Pu, Y., Liang, W., Hao, Y., Yuan, Y., Yang, Y., Zhang, C., Hu, H. and Huang, G.	Rank-DETR	COCO	Improved AP scores
DiffusionDet: Diffusion Model for Object Detection	Chen, S., Sun, P., Song, Y. and Luo, P.	DiffusionNet	COCO, CrowdHuman, LVIS	Obtained an AP of 46.8 with a ResNet-50 backbone. Outperforms other detectors on datasets.
PROB: Probabilistic Objectness for Open World Object Detection	Zohar, O., Wang, K.C. and Yeung, S.	PROB integrates transformer, probabilistic objectness	M-OWODB, S-OWODB benchmarks	2-3x U-recall improvement, mAP boost
CR-YOLOv8: Multiscale Object Detection in Traffic Sign Images	Zhang, L.J., Fang, J.J., Liu, Y.X., Le, H.F., Rao, Z.Q. and Zhao, J.X.	CR-YOLOv8	TT100k	CBAM for spatial and channel features; RFB for feature diversity; optimized loss function.
A multi-scale traffic object detection algorithm for road scenes based on improved YOLOv5	Li, A., Sun, S., Zhang, Z., Feng, M., Wu, C. and Li, W.	YOLOv5s model, incorporating CARAFE, SPD-Conv, and NAM	Real time	7.1% accuracy improvement on a diverse traffic scene dataset
A Robust Framework for Object Detection in a Traffic Surveillance System	Akhtar, M.J., Mahum, R., Butt, F.S., Amin, R., El-Sherbeeny, A.M., Lee, S.M. and Shaikh, S.	YOLOv2 enhanced with DenseNet-201	Kaggle Vehicle Dataset , KITTI Dataset, Pascal VOC Dataset, MS COCO Dataset	Improved result in all datasets
STC-YOLO: Small Object Detection Network for Traffic Signs in Complex Environments	Lai, H., Chen, L., Liu, W., Yan, Z. and Ye, S.	-	Enhanced TT100K	+9.3% mAP over YOLOv5
A small object detection algorithm for traffic signs based on improved YOLOv7	Li, S., Wang, S. and Wang, P.	SANO-YOLOv7	TT100K	88.7% mAP

Table 1: Summary and findings of the above research papers

when the servers are own that might put a risk towards jeopardizing the traffic solution.

3 Methodology

3.1 Dataset

Our goal in this project is to improve traffic optimisation by object identification using the KITTI dataset (Geiger et al., 2013), a well-known benchmark in the area of autonomous driving research. The KITTI collection features high-resolution car-camera images shot in a variety of environments, such as cities, countrysides, and highways.

Figure 2: Sample images of the Kitti dataset (Source: KITTI)

Car, Pedestrian, Van, Cyclist, Truck, Misc, Tram, and Person Sitting are the eight object types represented in the YOLO-formatted KITTI_YOLO_LABELS dataset. Separate sets of images, totaling 749, were used for training and testing. The training set had 6,732 shots. We divided up the work such that we could train and evaluate different versions of the YOLO model (YOLOv3, YOLOv4, YOLOv8, YOLOv9, YOLOv10) with the other models (SSDNet, RetinaNet, Mask R-CNN) to see how well they handled object detection in traffic.

3.2 SSDNet

SSDNet (Huang et al., 2023) is one of the most famous object detection models distinguished with remarkable velocity, as well as accuracy. One deep neural network is used to predict the coordinates of the bounding boxes and the probability of the classes of multiple items in a single pass through the network. This is based on the usage of fixed set of bounding boxes referred to as priors to predict the appropriate bounding boxes for objects of different sizes. Organised SSDNet is best used for real-time object detection mainly in the traffic optimization as quick and accurate results are required in this aspect.

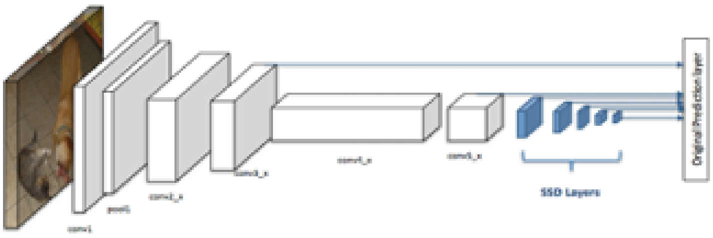


Figure 3: Architecture of SSDNET Huang et al. (2023)

The strength of this technology is that it is able to identify multiple objects within a single scan; therefore, it proves to be particularly useful and accurate when analyzing video feeds recorded by traffic cameras. This model helps in noticing automobiles, pedestrians and other items that can be of importance in fine tuning the traffic.

3.3 RetinaNET

RetinaNet (Li and Ren, 2019) is a monolithic identification model specifically acclaimed for its ability to judge objects of different sizes. The approach of the work harnesses a feature pyramid network and a focus loss function to balance between the foreground and background classes. RetinaNet, as a type of single stage detector, uses a primary network (such as ResNet), and then an FPN network to extract features of different scales.

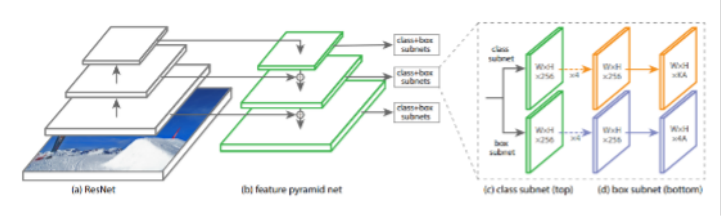


Figure 4: Retina model architecture Li and Ren (2019)

The specific architecture of RetinaNet's feature pyramid network (FPN) enables the identification of objects situated at varying scales, which is particularity advantageous for traffic scenes where objects of various sizes might be challenging to identify. This particular model can track and identify automobiles and pedestrians and other objects as accurately as in real life, thus creating valuable data that can help improve the traffic flow, and efficient safety measures.

3.4 Mask R-CNN (Mask Region-based Convolutional Neural Network)

As a result, segmentation masks for every occurrence can be predicted and thus, Mask R-CNN augmented the Faster R-CNN model (He et al., 2017). Its main application is instance segmentation, and the task of this network is to detect and segment objects at the pixel level. Mask prediction branch that is added within the Faster R-CNN framework is what makes it different from Mask R-CNN. As a next step, in the regional proposals, it uses a bounding-box regression network that is modified from a region proposal network (RPN).

Depending on the traffic scenes, it may be useful to use Mask R-CNN for instance segmentation for finer object analysis. This model is not related to traffic optimization but may give further information concerning shapes and size of the objects which can be useful when considered in the context of traffic and its possible behaviors.

3.5 YOLOv3

The third version of the ideologically titled YOLO , is Faster and More Accurate For Object Recognition Than The Earlier Versions Redmon and Farhadi (2018). Continuing from the feature maps, it takes the input picture and generates a grid and every cell gives

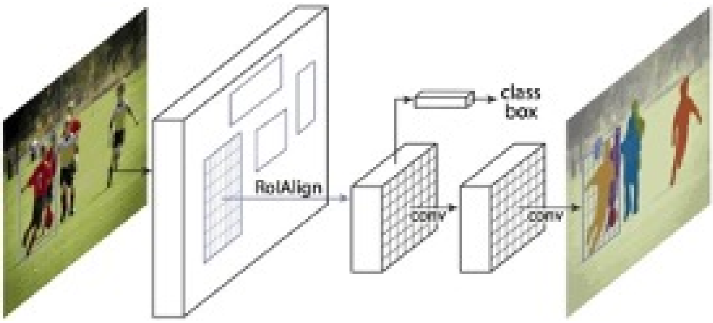


Figure 5: Mask R-CNN Framework (He et al., 2017)

class probability and bounding boxes. YOLOv3 is based on a Darknet-53 network; detection layers follow. Outcomes of different dimensions can be dealt by the help of bounding boxes predicted in three different scales. Since identification of car and pedestrians are crucial in real environment for the improvement of traffic flows in real-time, YOLOv3 can be useful in this work.

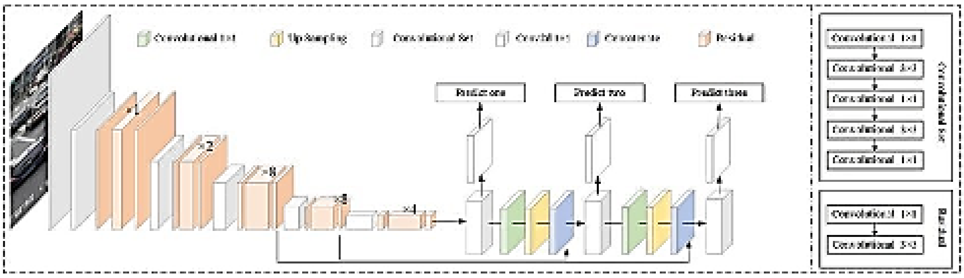


Figure 6: Architecture of YOLOv3 (Redmon and Farhadi, 2018)

3.6 Yolov4

YOLOv4 (Bochkovskiy et al., 2020) is an improved version of YOLOv3 created for higher efficiency with the use of new tools and tweaks. To acquire improved performance, it employs several architectural changes and training methods. YOLOv4 uses CSPDarknet53 backbone and adds a new feature fusion module called PANet. Besides, it makes use of an adjusted version of BiFPN for feature fusion purposes to form a pyramid structure. That is why YOLOv4's increased speed and reliability can produce even better results in traffic management and especially in complex traffic scenarios.

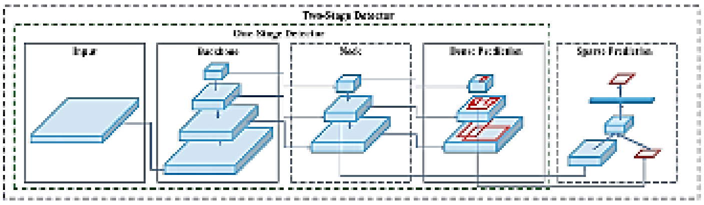


Figure 7: Architecture of YOLOv4 (Bochkovskiy et al., 2020)

The enhanced speed and accuracy of YOLOv4 make it a highly appealing option for optimising traffic-related operations. The capacity to perceive items at various magnitudes and in intricate surroundings can aid in the precise identification and monitoring of automobiles, pedestrians, and other entities. The performance improvements of YOLOv4 can boost the effectiveness of traffic control tactics, particularly in dynamic traffic scenarios.

3.7 Yolov8

YOLOv8 prioritises efficiency without compromising accuracy (Sohan et al., 2024). The system employs a specialised and efficient framework that incorporates a lightweight backbone with grouped convolutions and optimised pointwise operations to minimise computing expenses. If our requirement is real-time processing and has resource limits, such as deploying the model on edge devices, YOLOv8's emphasis on efficiency can be beneficial.

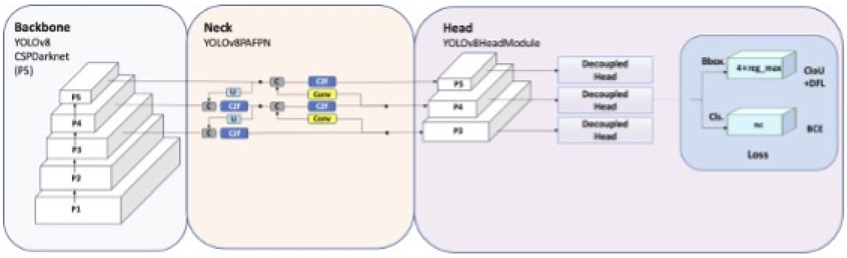


Figure 8: YOLOv8 architecture (Sohan et al., 2024)

3.8 YOLOv9

YOLOv9 enhances the existing framework of YOLOv8 by integrating vision transformers (ViTs) to effectively capture long-range relationships in images (Wang, Yeh and Liao, 2024). There is a possibility that this can enhance the way features are represented and improve the accuracy of object detection. Nevertheless, ViTs have the potential to escalate computational complexity.

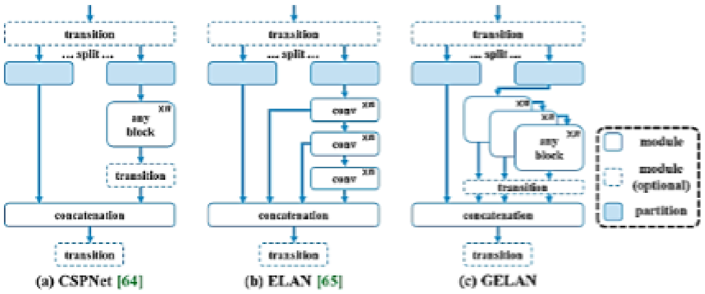


Figure 9: YOLOv9 architecture using CSPNet, ELAN and GELAN (Wang, Yeh and Liao, 2024)

3.9 YOLOv10

During training, YOLOv10 Wang, Chen, Liu, Chen, Lin, Han and Ding (2024) implements a new dual-head architecture that can handle one-to-many and one-to-one object assignments. Thanks to this breakthrough, post-processing Non-Maximum Suppression (NMS) is no longer necessary, which speeds up prediction and cuts down on latency.

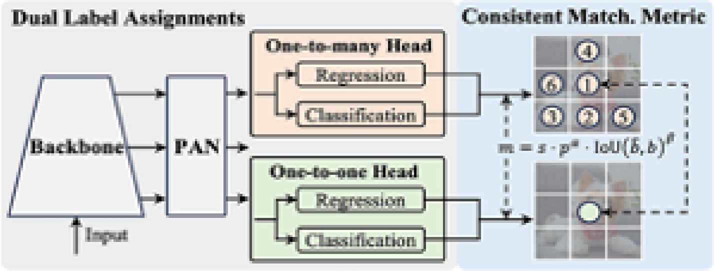


Figure 10: consistent dual assignment for NMS free training (Wang, Chen, Liu, Chen, Lin, Han and Ding, 2024)

A major benefit of YOLOv10's NMS-free training is its ability to minimise inference latency, which is an important consideration for many projects. The detection accuracy could be enhanced by the dual-head architecture as well. If our main aim in project places greater importance on achieving accuracy rather than speed and has the computational resources, utilising ViTs in YOLOv9 could prove advantageous for analysing intricate traffic situations.

3.10 Evaluation

Mean Average Precision (mAP) is a quantitative measure employed to assess the performance of object identification models, including Fast R-CNN, YOLO, Mask R-CNN, and others. The average precision (AP) values are computed by taking the mean of the accuracy values at different recall levels ranging from 0 to 1. Here is the formula for calculating the mAP:

$$mAP = \frac{1}{N} * \sum AP(c)|$$

Where, N is the total number of classes and AP(c) is the Average Precision for class c.

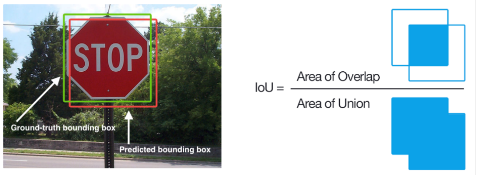


Figure 11: IOU and mAP representation (Source: TowardsDatascience)

Intersection over Union measures the extent to which the anticipated bounding box coordinates overlap with the ground truth box. A higher IoU value implies a close resemblance between the anticipated bounding box coordinates and the ground truth box coordinates.

4 Design Specification and Implementation

The below figure demonstrates a pipeline to be used for Object detection in vehicle dataset using best of the models (YOLOv10 and YOLOv8) that requires a methodical approach to improving detection performance by utilising different images either in the standard datasets or in the real time datasets.

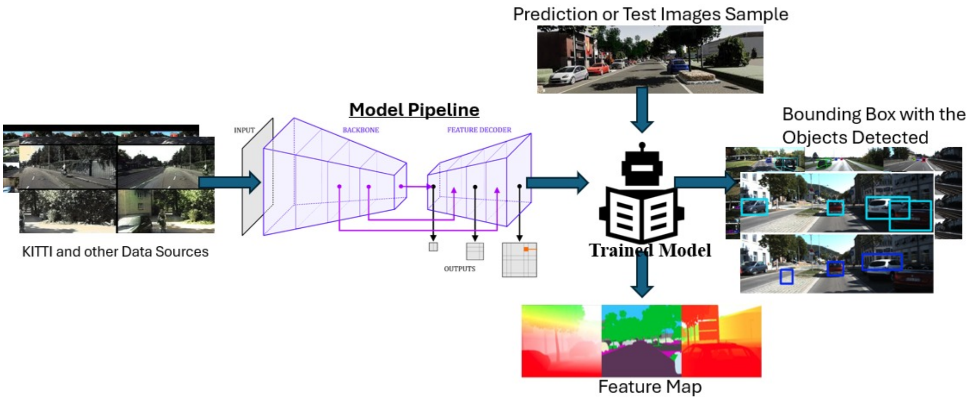

Prediction or Test Images Sample
Model Pipeline
INPUT
BACKBONE
FEATURE DECODER
OUTPUTS
Bounding Box with the Objects Detected
KITTI and other Data Sources
Trained Model
Feature Map

Figure 12: Flow diagram for implementing Object Detection using YOLO

Algorithm Flow:

Step 1 – Data Collection: Collect a diverse range of traffic data like KITTI, COCO or TT100K (after thorough research analysis) capturing the different traffic objects at various time, ensuring representation under various lighting conditions and scenarios.

Step 2 – Data Preprocessing: Data preprocessing include resizing and augmenting RGB images with pixel values normalised to [0, 1], and log transformation and normalisation for images. Custom data loaders are implemented, datasets are partitioned, and annotations are translated to different object detection models and YOLO format. In order to choose the best priors for the bounding boxes, k-means clustering is used to find the anchor boxes.

Step 3 - Annotation Parsing: Coordinate Adjustment: Parse manually annotated object locations, adjusting coordinates based on image dimensions to ensure alignment with model predictions. Formatting: Format parsed annotations into lists for fusion with model predictions.

Step 4 - Model Training: Separate Processing: Process RGB and IR images independently using different models to predict object locations in each modality, capturing unique features present in both spectra. Conversion to Lists: Convert the model predictions, comprising object bounding boxes and confidence scores, into list formats for subsequent fusion.

5 Evaluation

5.1 Case 1: Using Non – YOLO Models

5.1.1 SSDNET

An evaluation was conducted using the KITTI dataset to assess the performance of the SSDNet. The results revealed differing levels of success for various object types. The detector exhibited excellent performance in accurately detecting automobiles, which are frequently observed on roadways. Nevertheless, it encountered difficulties in accurately identifying bicycles, particularly when they were positioned in front or behind the vehicle. The detector encountered difficulties in recognising items of small size, objects with low contrast or luminance, and heavy-duty vehicles. Although the SSDNet has a pretty quick detection time, it has limitations in terms of its detection accuracy and the number of instances it can detect per image. The findings emphasise the trade-offs that exist between the speed and accuracy of object detection algorithms, and the potential impact this has on practical applications like traffic optimisation.

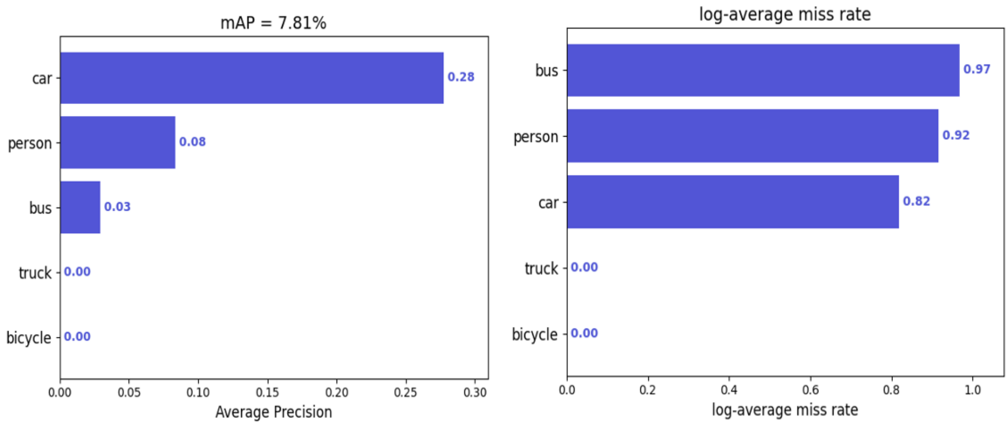


Figure 13: mAP on validation images and Logarithmic miss rate per class for ssdnet

Our observation indicates that the SSD net's ability to detect bicycles is unsatisfactory when the image shows the front or back view of the bicycles. However, its performance is comparatively better when the image shows the side view of the bicycles. Since vehicles are the most often observed things on roadways, the dataset images contain a large number of objects classified as "car". As a result, the average precision (AP) for the "car" class is the highest. Furthermore, the proportion of accurate positive predictions is at its highest for the category labelled as "car".

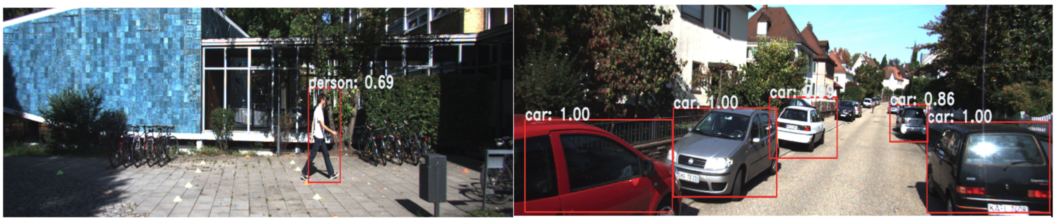


Figure 14: Predictions of ssdnet on validation data

Although the network can identify the car even when it is partially visible or in different orientations, it is unable to do so when the car's pixel size is very small. As evident from the image provided, the SSD Net fails to recognise many of the cars that are situated far away from the camera.

5.1.2 Mask Region-based Convolutional Neural Networks (Mask RCNN)

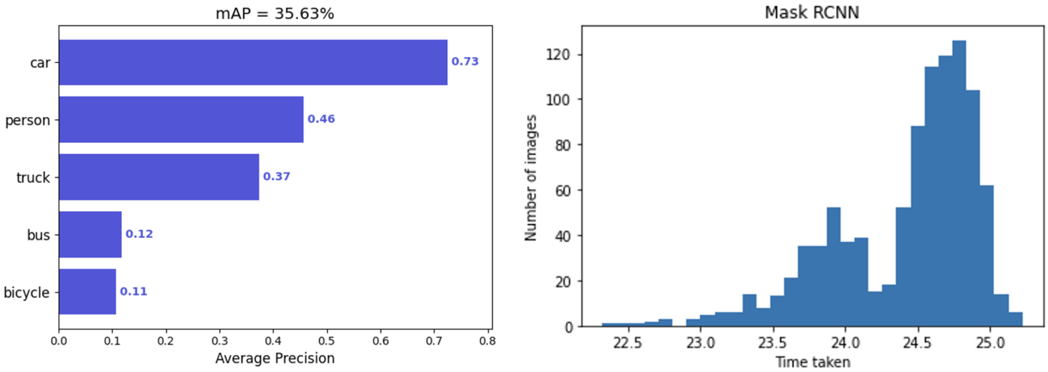


Figure 15: mAP for Mask – RCNN and The Histogram of the number of images detected within given range of time

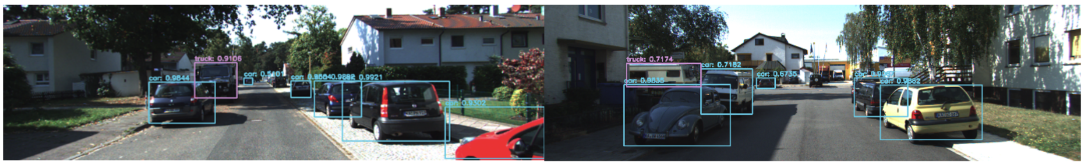


Figure 16: Mask – RCNN prediction bounding boxes on the testing samples

The model's capacity to accurately shade pixels belonging to the item facilitated precise predictions of the bounding box. One major limitation of Mask R-CNN is its long detection time per image, which averages around 24 seconds for all potential items in an image.Despite its slower speed compared to SSD Net and YOLOv3, Mask R-CNN exhibited superior detection accuracy and a greater number of instances detected per image.

5.1.3 RetinaNet

The ”car” class has the highest AP among the six classes. 2.The detection of buses and trains is relatively inadequate. The accuracy in identifying trucks is superior to that of buses and trains. RetinaNet demonstrates superior detection accuracy for bicycles in comparison to SSDNet and YOLOv3. By incorporating two distinct bicycle classes, namely stationary and non-stationary, the model enhances its ability to effectively learn and distinguish bicycles in different situations, resulting in improved performance.

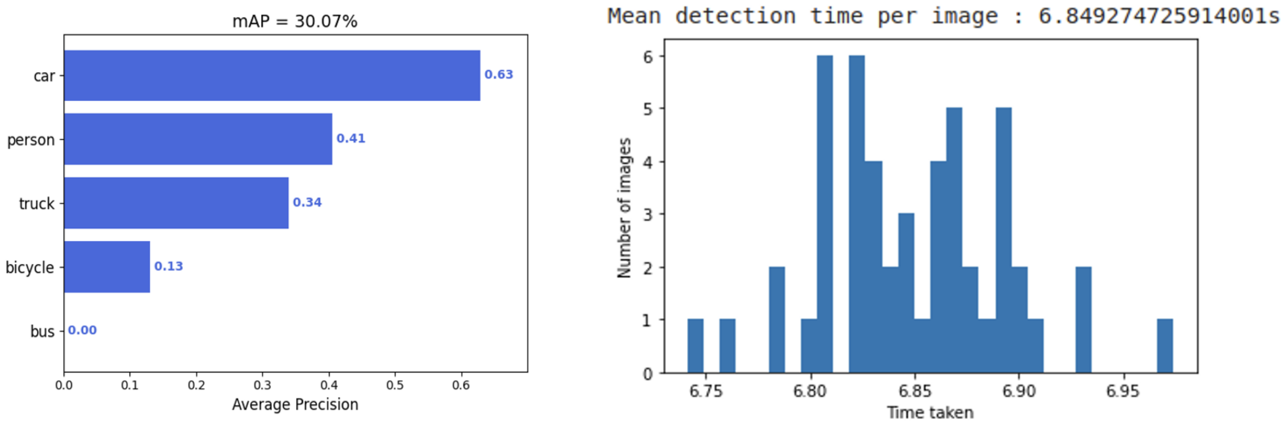


Figure 17: The Histogram of the number of images detected within given range of time

RetinaNet is highly efficient at detecting individuals, even when they are present in a limited portion of the image’s pixel space. RetinaNet encounters difficulties in accurately identifying objects in conditions of insufficient lighting and obstruction. The average detection time for RetinaNet is roughly 6.9 seconds per image. While this method is quicker than Mask R-CNN and similar in speed to YOLOv4, it is not as fast as SSDNet and YOLOv3.

5.2 Case 2: Using YOLO Models

5.2.1 Yolov3

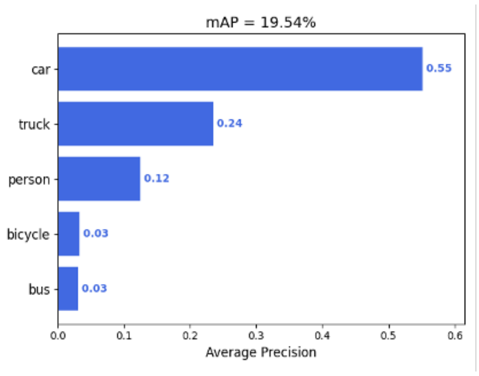


Figure 18: mAP for yolov3

The performance differed among object classes in the KITTI dataset. The system performed well in detecting cars, buses, trucks, and trains,person but struggled with bicycles, especially when they were viewed from the front or rear.

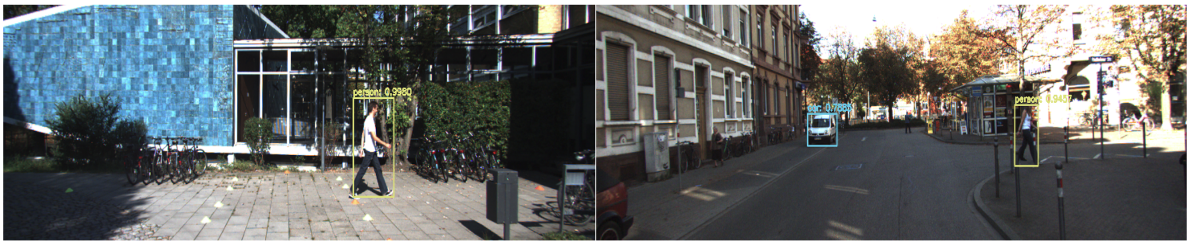


Figure 19: YOLOv3 prediction

The ratio of correctly detected objects and the number of objects diffused in the given image were fairly high. In particular, the average of the detection time has been revealed to be around 4. It successfully achieved 49 seconds per image on thousand of KITTI dataset images. The system gave better results than SSD Net in aspects of object detection and in predicting the coordinates of the bounding box. It mentioned that the system successfully recognized objects of interest and could do so even when the objects are partially hidden.

5.2.2 YOLOv4

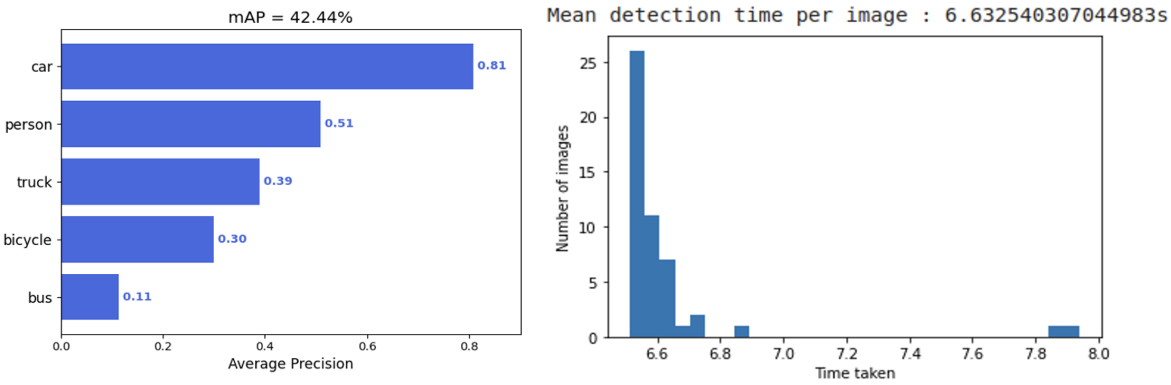


Figure 20: The Histogram of the number of images detected within given range of time and The Histogram of the number of images detected within given range of time

The detection accuracy for buses was superior in comparison to other models, whereas the performance for trains was extremely inadequate. The model exhibited enhanced efficacy in identifying heavy-duty vehicles when compared to YOLOv3, particularly in the detection of trucks.

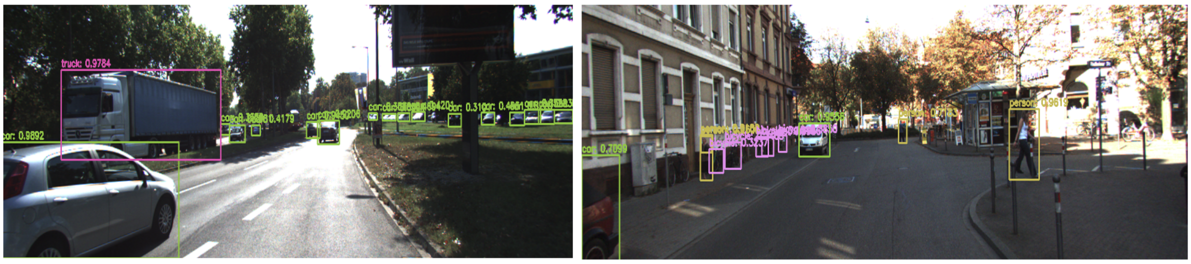


Figure 21: Ability of the model YOLOv4 to detect objects of small size and occupying small area in the image's pixel space

While using YOLOv4, all the problems noted in the previous version, namely YOLOv3, were eliminated, including the identification of items that are away from the camera or occupy a small part of the frame. As a result, this model proved that it excelled at defining small things that occupy a small area of the image pixel space. Thus, compared to other three models above which don't belong to YOLO family, YOLOv4 showed a higher accuracy in the identification of the objects belonging to the heavy-duty vehicle class. The work demonstrated better result in object detection especially where there was low contrast. Another disadvantage of the YOLOv4 is the relatively long time it takes for the model to make the detection on each of the input images even though it gave good results when it came to the detection of objects. This might lead to a situation where the identification of objects in every frame in streaming applications will be done very much later. The model improved in terms of its ability to detect bicycles within images and more so when the bicycles were side lit. The YOLOv4 model proved to have the average of about 6 mean detection time. 6 seconds per image when trained and tested on the first 50 images of the inherently challenging KITTI dataset. The detection time is rather short compared to other models that were discussed in this paper, and the acceptable accuracy of detections is achieved.

5.3 Case 3: Using Latest YOLO Models

5.3.1 YOLOv8:

In overall, from detected on the roads YOLOv8 shows a great result of recognizing cars in all scenarios. This model places rectangular frames around multiple cars, along with probabilities of the detections made by the model. What is also interesting is that the confidence scores also vary concerning the detections. Some detections will have scores that are close to 1. Thus, the first candidate for the detected object is a car with the recognition level of 0, meaning that there is an extremely high degree of confidence that it is indeed a car. lower values in other cases suggest that the level of absolute confidence in the given labels suggested by the model is less. Most of the bounding boxes ideally should enclose automobiles but it is possible that some may contain other objects that might have been mistaken by the model.

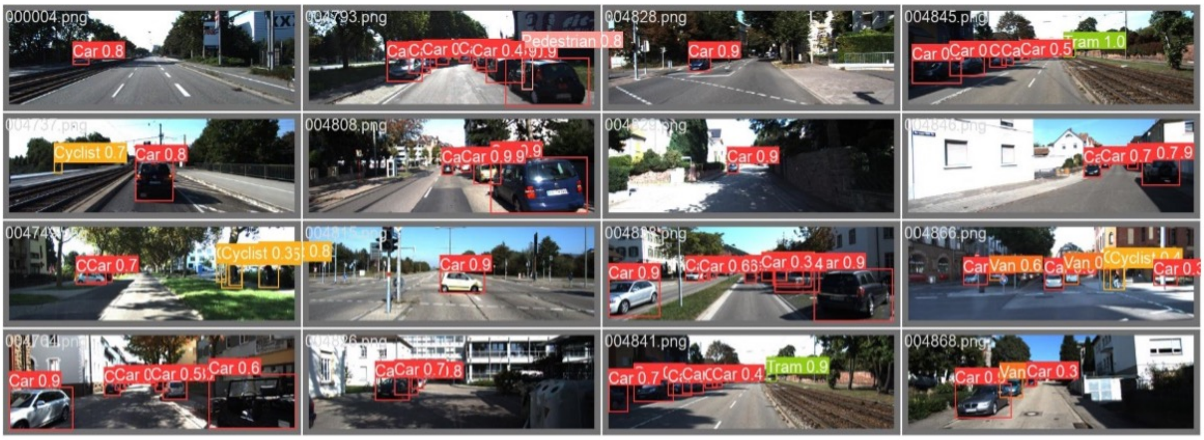


Figure 22: YOLOv8 near to accurate predictions

Thus, moving objects that have similar shapes to vehicles, for instance, SUVs or vans parked along a slope may be grouped in the wrong way. In relation to this particular image, the bounding box detected at the bottom left can be an object such as the traffic cone or small bush as opposed to a car as it has a mere confidence score of 0. 7.

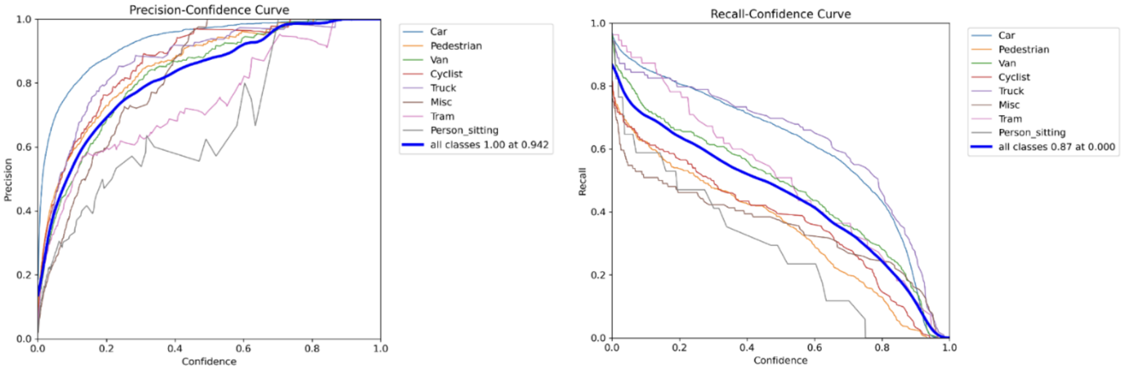
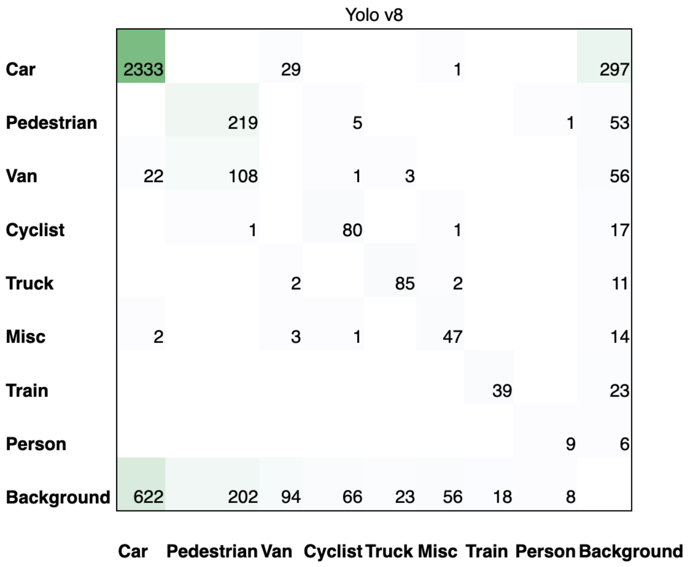


Figure 23: Precision and recall curves for yolov8

The graph demonstrates that when the confidence threshold increases (approaching 1.0 on the X-axis), YOLOv8 exhibits a high level of precision (approaching 1.0 on the Y-axis). This indicates that when the model attributes a high level of certainty to a detection, there is a strong probability that it is indeed a correct car.



	Car	Pedestrian	Van	Cyclist	Truck	Misc	Train	Person	Background
Car	2333		29			1			297
Pedestrian		219		5				1	53
Van	22	108		1	3				56
Cyclist		1		80		1			17
Truck			2		85	2			11
Misc	2		3	1		47			14
Train							39		23
Person								9	6
Background	622	202	94	66	23	56	18	8	

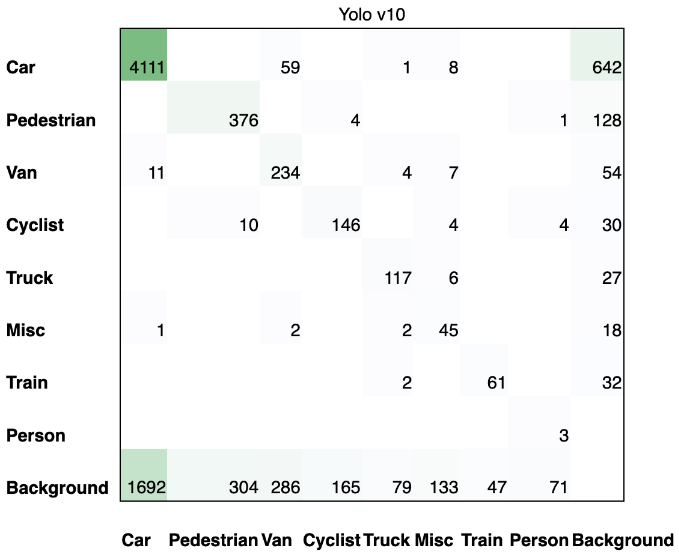
Figure 24: Confusion matrix for YOLOv8

Lowering of the confidence threshold means going to the left on the X-axis scale, and it corresponds to going down on the Y-axis scale in terms of the precision. This implies that while YOLOv8 performs a better recall in objects' detection, it also tends mistakenly tag things with low confidence scores that are in fact erroneous.

5.3.2 5.3.2 YOLOv9

The YOLOv9 didn't perform well since the model took huge amount of time for the prediction and hence, we ruled it out from the performance evaluation.

5.3.3 5.3.3 YOLOv10



	Car	Pedestrian	Van	Cyclist	Truck	Misc	Train	Person	Background
Car	4111		59		1	8			642
Pedestrian		376		4				1	128
Van	11		234		4	7			54
Cyclist		10		146		4		4	30
Truck					117	6			27
Misc	1		2		2	45			18
Train					2		61		32
Person								3	
Background	1692	304	286	165	79	133	47	71	

Figure 25: Confusion matrix for YOLOv10

As seen from the above figure, like v8, v10 exhibits the similar performance in the detection of the cars and a high positive rate (PR) is seen in the confusion matrix. As we deep dive into the prediction of the test images, the model tries to identifying the feature maps in better way for each individual objects.

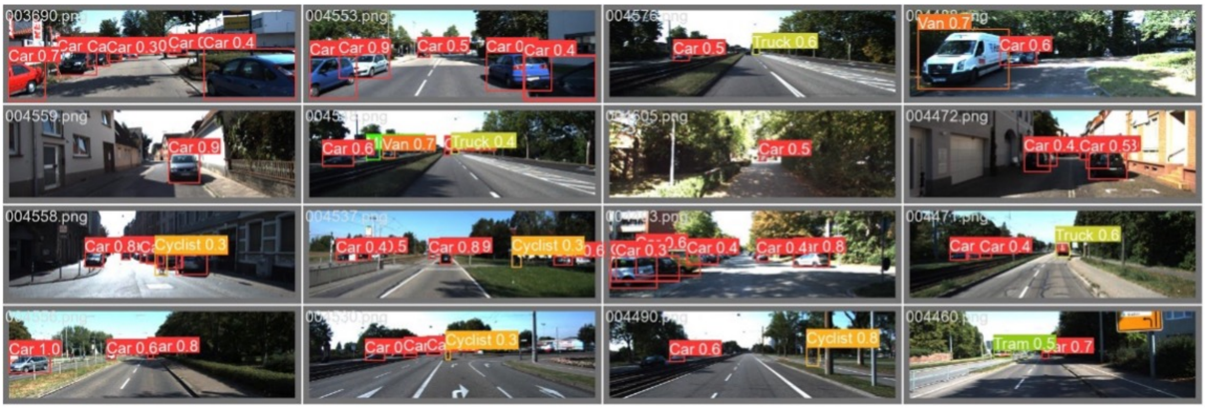


Figure 26: YOLOv10 predictions on validation data

When aligned to the corresponding bounding boxes this assists in the identification of the objects appropriately. Now this influences in the terms of computations of the model performance. Besides, even for the class of pedestrians YOLOv10 could recognize, the confusion matrix means that there could be many cases unobserved or classified wrongly.

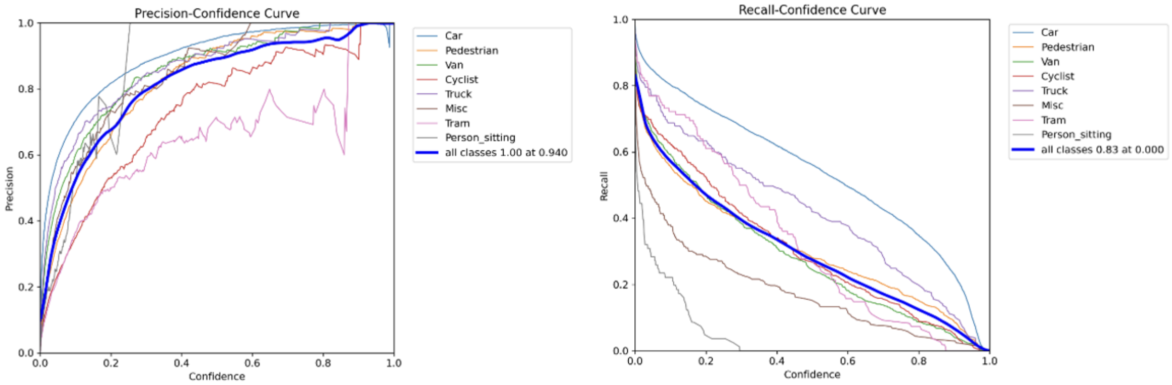


Figure 27: Precision and recall curves for YOLOv10

5.4 Discussion

Cars : Both YOLOv10 and YOLOv8 demonstrated excellent precision (greater than 80%) in detecting vehicles, indicating successful recognition of the majority of car instances. YOLOv10 had a marginal recall advantage (91% vs 86%), indicating that it detected fewer automobiles compared to YOLOv8. Other models didn't perform well against the YOLO models. YOLOv4 also performed well.

Pedestrian Detection : Neither model performed as well when it came to detecting pedestrians as it did when it came to cars. YOLOv10 outperformed YOLOv8 in terms

mAP	SSDNet	MRCNN	RetNet	Yv3	Yv4	Yv8	Yv9	Yv10
car	28%	73%	63%	55%	81%	89%	8%	88%
Person	8%	46%	41%	12%	51%	68%	11%	71%
Truck	0%	37%	34%	24%	39%	72%	5%	60%
Bicycle	0%	32%	13%	3%	30%	63%	3%	72%
Bus	3%	11%	0%	3%	11%	52%	2%	52%

Table 2: mAP comparison for different objects for different models

Class	YOLOv8 Precision	YOLOv8 Recall	YOLOv10 Precision	YOLOv10 Recall
car	88%	83%	84%	91%
Padestrian	81%	63%	89%	75%
Van	89%	77%	92%	7%
Cyclist	65%	88%	93%	62%

Table 3: precision and recall comparison for YOLOv8 and YOLOv10

of accuracy, reaching 89% compared to 81%. Yet, YOLOv10 failed to detect a higher number of pedestrians (25% vs. 37%).

Other classes : The detection of Van, Truck, Tram, Misc, and Person sitting classes was affected by both models to a lesser extent. The recall for both models stayed low, indicating a considerable proportion of missed detections, even though YOLOv10 generally exhibited superior precision for these classes.

mAP	SSDNet	MRCNN	RetNet	Yv3	Yv4	Yv8	Yv9	Yv10
Mean Average Time(s)	8.34	24.09	6.84	4.49	6.6	5.32	62.6	7.89

Table 4: Computation Time comparison

Comparing the two models, it can be reckon that YOLOv10 and YOLOv8 took less time on average, for each model and it showed a higher overall accuracy in completely identify the frame content especially the cars and pedestrians which are very vital for traffic efficiency as shown in the diagram below that demonstrated the greatest overall accuracy in the important classes including automobiles and people. Although, these models seem to have some limitations in handling some object classes that could limit their adaption to real world usage. As can be observed from the analysis made in the course of the investigation, it can wager that in general, YOLO has better performance. Difficult courses, it was always a bother to consistently accurately identify bicycles of any make and model, even from anterior or posterior views. The presented YOLO models showed higher speed of detections, and therefore are more effective to be implemented in real-time conditions. While Mask R-CNN was very accurate in its detection, it recorded the slowest time to effect the detection, meaning it is unsuitable for accurate detection in real-time traffic flow. Based on the experiment results, evidently, YOLOv8 achieved better results in detecting small objects and things that can be contained in a limited portion of the image, greater than the YOLOv3 and SSDNet. In the case of detecting trucks and trains, YOLOv4 and Mask R-CNN were leading the pack regarding the detection efficiency. It is seen that by merging two different groups namely stationary and non-stationary groups, the system provided a very high degree of accuracy in the identification of bicycles. However, it had complications in recognizing objects in conditions of low light and when there were objects blocking view and also had problems in distinguishing between numerous bicycles placed close to each other.

6 Conclusion and Future Work

Finally, it can be stated that YOLOv8 and YOLOv10 are generally more effective in detecting vehicles and people, which are important for increasing the traffic speed. Observing the results for YOLOv10 the model demonstrates higher overall precision as well as the recall rate for automobiles and pedestrians in contrast to YOLOv8, which has 81% accuracy for the detection of pedestrians. Specifically, the evaluation indicated that the YOLOv10 design attained a 89 % accuracy rate specifically for pedestrians. Some issues are observed when distinguishing between categories; it is worse in some cases, namely with vans, trucks, trams, diverse items, and seated people as well as cases when the number of detections is higher than the number of actual objects and recall rates are lower. On the other hand, it has to be mentioned that both the models have some drawbacks. Nonetheless, bicycles remain rather a strenuous asset particularly when seen from the forward or backward position of the car. As for the speed, YOLO models are more appropriate for real-time applications due to shorter detection time. Within the same regard, the Mask R-CNN models but requires more time to detect, thus being slightly less appropriate for real-time applications. From the results, it can be concluded that YOLOv8 algorithm is again outperforming YOLOv3 and SSDNet algorithms in detecting small objects and objects that are enclosed in a particular region of an image. YOLOv4 and Mask R-CNN algorithms are proved to be superior in efficiency as far as detecting robust vehicles like trains and trucks. Even though the applications of stationary and non-stationary detection technologies enhance the bicycle identifying chances, issues arise at night or in areas with hindrances. In general, it can be concluded that YOLO models are perfect for real-time traffic due to the balance achieved between accuracy and time. But here it is suggested to make additional enhancements to the suggested method in case if the object classes are rather difficult.

Certain classes such as bicycles, vans, trucks, trams, and persons sitting should be the focus of assessment for future works in environments with low light or where the visibility is restricted in some way. Other such approaches are the Hybrid models like YOLOv3 that offers the speed of YOLOv3 along with the accuracy of YOLOv10 would be suited for real-time use. This would make it possible to obtain high efficiency of operating the supply chain network. It would also be possible to perform proper time-critical processing in cases of limited resources by optimizing models for distribution within the edge devices. It is significant to perform a critical analysis of the honest-world issues for bolstering the model's proficiency in the real world, and to integrate the model with the present-day traffic controlling systems. Such improvements may be obtained by going deeper into new architectures of AI and utilizing highly complex post-processing techniques for enhancing the creative detection skills. Among the beneficial effects of this new traffic flow network, it will be in part possible to obtain a higher efficiency and traffic safety.

References

Akhtar, M. J., Mahum, R., Butt, F. S., Amin, R., El-Sherbeeny, A. M., Lee, S. M. and Shaikh, S. (2022). A robust framework for object detection in a traffic surveillance system, *Electronics* **11**(21): 3425.

Barbosa, R., Ogobuchi, O. D., Joy, O. O., Saadi, M., Rosa, R. L., Al Otaibi, S. and Rodríguez, D. Z. (2023). Iot based real-time traffic monitoring system using images sensors by sparse deep learning algorithm, *Computer Communications* **210**: 321–330.

Bochkovskiy, A., Wang, C.-Y. and Liao, H.-Y. M. (2020). Yolov4: Optimal speed and accuracy of object detection, *arXiv preprint arXiv:2004.10934* .

Chen, S., Sun, P., Song, Y. and Luo, P. (2023). Diffusiondet: Diffusion model for object detection, *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 19830–19843.

Geiger, A., Lenz, P., Stiller, C. and Urtasun, R. (2013). Vision meets robotics: The kitti dataset, *The International Journal of Robotics Research* **32**(11): 1231–1237.

He, K., Gkioxari, G., Dollár, P. and Girshick, R. (2017). Mask r-cnn, *Proceedings of the IEEE international conference on computer vision*, pp. 2961–2969.

Huang, D., Zhang, Q., Wen, Z., Hu, M. and Xu, W. (2023). Research on a time series data prediction model based on causal feature weight adjustment, *Applied Sciences* **13**(19): 10782.

Iftikhar, S., Asim, M., Zhang, Z., Muthanna, A., Chen, J., El-Affendi, M., Sedik, A. and Abd El-Latif, A. A. (2023). Target detection and recognition for traffic congestion in smart cities using deep learning-enabled uavs: A review and analysis, *Applied Sciences* **13**(6): 3995.

Lai, H., Chen, L., Liu, W., Yan, Z. and Ye, S. (2023). Stc-yolo: small object detection network for traffic signs in complex environments, *Sensors* **23**(11): 5307.

Li, A., Sun, S., Zhang, Z., Feng, M., Wu, C. and Li, W. (2023). A multi-scale traffic object detection algorithm for road scenes based on improved yolov5, *Electronics* **12**(4): 878.

Li, S., Wang, S. and Wang, P. (2023). A small object detection algorithm for traffic signs based on improved yolov7, *Sensors* **23**(16): 7145.

Li, Y. and Ren, F. (2019). Light-weight retinanet for object detection, *arXiv preprint arXiv:1905.10011* .

Madhavi, G. B., Bhavani, A. D., Reddy, Y. S., Kiran, A., Chitra, N. T. and Reddy, P. C. S. (2023). Traffic congestion detection from surveillance videos using deep learning, *2023 International Conference on Computer, Electronics & Electrical Engineering & their Applications (IC2E3)*, IEEE, pp. 1–5.

Ng, S.-C. and Kwok, C.-P. (2020). An intelligent traffic light system using object detection and evolutionary algorithm for alleviating traffic congestion in hong kong, *International journal of computational intelligence systems* **13**(1): 802–809.

Pu, Y., Liang, W., Hao, Y., Yuan, Y., Yang, Y., Zhang, C., Hu, H. and Huang, G. (2024). Rank-detr for high quality object detection, *Advances in Neural Information Processing Systems* **36**.

Ramana, K., Srivastava, G., Kumar, M. R., Gadekallu, T. R., Lin, J. C.-W., Alazab, M. and Iwendi, C. (2023). A vision transformer approach for traffic congestion prediction in urban areas, *IEEE Transactions on Intelligent Transportation Systems* **24**(4): 3922–3934.

Redmon, J. and Farhadi, A. (2018). Yolov3: An incremental improvement, *arXiv preprint arXiv:1804.02767* .

SenthilPandi, S., Paulraj, D., Mithun, D. and Kumar, N. (2023). Object detection using learning algorithm and iot, *2023 International Conference on Research Methodologies in Knowledge Management, Artificial Intelligence and Telecommunication Engineering (RMKMATE)*, IEEE, pp. 1–6.

Sohan, M., Sai Ram, T., Reddy, R. and Venkata, C. (2024). A review on yolov8 and its advancements, *International Conference on Data Intelligence and Cognitive Informatics*, Springer, pp. 529–545.

Wang, A., Chen, H., Liu, L., Chen, K., Lin, Z., Han, J. and Ding, G. (2024). Yolov10: Real-time end-to-end object detection, *arXiv preprint arXiv:2405.14458* .

Wang, C., Chen, Y., Wang, J. and Qian, J. (2023). An improved crowddet algorithm for traffic congestion detection in expressway scenarios, *Applied Sciences* **13**(12): 7174.

Wang, C.-Y., Yeh, I.-H. and Liao, H.-Y. M. (2024). Yolov9: Learning what you want to learn using programmable gradient information, *arXiv preprint arXiv:2402.13616* .

Zhang, L. J., Fang, J. J., Liu, Y. X., Le, H. F., Rao, Z. Q. and Zhao, J. X. (2023). Cr-yolov8: Multiscale object detection in traffic sign images, *IEEE Access* **12**: 219–228.

Zhao, Y., Lv, W., Xu, S., Wei, J., Wang, G., Dang, Q., Liu, Y. and Chen, J. (2024). Detrs beat yolos on real-time object detection, *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 16965–16974.

Zohar, O., Wang, K.-C. and Yeung, S. (2023). Prob: Probabilistic objectness for open world object detection, *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11444–11453.