Wrong-Way Vehicle Detection Using YOLOv7 for Enhanced Traffic Safety

MSc Research Project
Data Analytics

Alphons Zacharia James
Student ID: x23169702

School of Computing
National College of Ireland

Supervisor: Furqan Rustam

National College of Ireland
Project Submission Sheet
School of Computing

Student Name:	Alphons Zacharia James
Student ID:	x23169702
Programme:	Data Analytics
Year:	Jan 24-25
Module:	MSc Research Project
Supervisor:	Furqan Rustam
Submission Due Date:	12/12/2024
Project Title:	Wrong-Way Vehicle Detection Using YOLOv7 for Enhanced Traffic Safety
Word Count:	XXX
Page Count:	23

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:	Alphons Zacharia James
Date:	29th January 2025

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):	

Wrong-Way Vehicle Detection Using YOLOv7 for Enhanced Traffic Safety

Alphons Zacharia James
x23169702

Abstract

Wrong-way driving is a significant contributor to road accidents and traffic congestion worldwide. Traditional methods for detecting wrong-way vehicles, such as manual monitoring, fixed cameras, or traffic sensors, have limitations in terms of real-time detection, scalability, and accuracy. These traditional systems often fail to provide timely alerts, especially in dynamic traffic conditions. This study addresses these gaps by implementing an advanced vehicle detection system using YOLOv7, which can accurately identify wrong-way drivers in real-time. So, the dataset used for this study is the "Vehicle Detection" dataset, which contains a diverse set of images representing different vehicle types, including ambulances, buses, cars, motorcycles, trucks, and vans. The primary objective of this study is to develop a robust vehicle detection system capable of identifying and tracking vehicles in real-time video streams. Several models were explored for this task, including YOLOv5, YOLOv8, and YOLOv7, with each model trained and tested on the vehicle dataset. Among these, the best performance was achieved using YOLOv7, which demonstrated the highest mAP@0.50 score ie 0.876, making it the optimal model for this vehicle detection task. YOLOv7 outperformed other models in terms of accuracy and precision, particularly excelling in detecting various vehicle classes, such as cars and trucks, with higher precision and recall values.The study also included the implementation of a real-time detection system that tracks vehicles and identifies wrong-way driving violations using a reference direction vector and the system is evaluated based on assumptions using sample traffic videos.

1 Introduction

1.1 Background

Urban traffic management faces various challenges including traffic flow management that must ensure the safety of motorists and pedestrians and has to ensure strict enforcement of traffic rules. The unidirectional traffic flow is set in certain roads for controlling traffic flow management thereby reducing traffic congestion. The violations of these restrictions can raise safety concerns for other motorists and even for pedestrians. Wrong-Way Vehicle Detection means a system aimed at detecting vehicles moving in the wrong direction on roads, highways or ramps, which is rather dangerous(Reddy et al.; 2024). Machine learning models can provide solutions for implementation of violation detection systems which can be more effective than manual surveillance monitoring. Deep learning models can provide more robust and accurate real time vehicle detection(Li et al.; 2020). It can be used in traffic management systems or implemented in smart cities to minimize the frequency of wrong way driving accidents, collision, disruption of traffic and associated fatalities. Such systems are gradually becoming required tools for contemporary traffic control.

1.2 Motivation

Wrong way driving can lead to accidents and can be even fatal if head on collisions happens at a high speed. Also, the existing works have shown promising advancement using machine learning and computer vision. From Redmon et al. (2016) study showed the effectiveness of YOLO in real time object detection making it suitable for detecting vehicles in wrong way vehicle detection system. Ren et al. (2015) introduced Faster R-CNN which is a deep learning model used for vehicle detection. These studies highlight the potential of machine learning techniques in wrong way detection system and enable automated solutions .The selection of the topic is driven by the need to build on these advancement and integrate machine learning methodologies to enhance road safety .

1.3 Research Question and objective

The research questions for this research is "How can YOLO an efficient deep learning model be optimized and implemented to accurately detect and track wrong-way vehicle movements in real-time?" and our objective is to design and implement a real-time system for detecting wrong-way vehicle movements on roadways using the YOLOv7 object detection model.

1.4 Research Problem

Vehicle travel in the wrong direction is one of the clear and present dangers to the lives and property of individuals on the road. Identifying such violations in real time is also difficult because of the unpredictability of traffic and road conditions as well as differences in vehicle models. The conventional surveillance systems do not have the intelligence and specificity needed to differentiate between wrong-way instances, thus typically causing delay or sending false alarms. As for today, with the development of deep learning and further object detection models such as YOLOv7 there is a prospect to solve this crucial problem. In its turn, YOLOv7 characterized by speed and high accuracy can be fine-tuned for vehicle detection and recognition of moving objects from the video stream. Also, incorporation of tracking techniques for assigning vehicle identification numbers and vector analysis for violation identification require wear-resistant solutions. This raises the need for a dedicated solution since currently, no scalable and accurate system that may perform well in various scenarios is available. This study seeks to implement a real-time wrong-way driving detection system under the YOLOv7, which will be effective, efficient and functional to prevent accidents on the roads.

1.5 Structure of the Report

This report is structured as follows: Introduction section explains the background to the study, research questions and objectives and research problem. Next section section literature review of other vehicle detection and tracking methods with an emphasis on

YOLO models.Next section research methodology explains the processes involved in the research,Design Specification section explains the architecture of the proposed system. The practical implementation of the system using YOLO model is explained in implementation part.The results of the study are evaluated in the following section and later sections explains conclusion ,future work and limitations.

2 Related Work

Traffic violation and wrong-way detection system are important in improving road safety and reducing the frequency of accidents .Different methods are used for this from traditional methods to advanced deep learning and IoT enabled systems are used for monitor traffic and detect the violations .This review focuses on methodologies of different approaches while highlighting their limitations and identifying gaps in existing literature.

2.1 Traffic Violation Detection System

Traffic violations are one of the major problems in the management of urban traffic. In order to ensure road safety and compliance with traffic law, new solutions must be developed to handle traffic violations.El Atigh and Özer (2021) outline a full-scale survey on traffic violation detection systems, including varied approaches that have been enforced globally. Their work constitutes a change from traditional surveillance towards more automated technology-driven solutions. Such development finds application in the case of one-way street violations; for such a violation is only effective if detected in real-time to deter further instances of non-compliance.

Over time, many systems have been developed to detect traffic violations. The systems range from simple monitoring using human senses to complex, automated systems. According to Akib et al. (2023) integrated systems are significant for efficient monitoring of highways' traffic, as they proved in their work focusing on the highways of Bangladesh. Even though this system was set up for highways, the principles and methodologies it uses to detect violations can be adopted for urban settings and modified to fit one-way street enforcement.Traffic violations detection has evolved from Manuel methods to semi-automated techniques.Xu et al. (2020)addressed this issue by combining ViBe technique with motion estimation methods although it improved accuracy of detection under varying illumination but it struggled with densely populated scenarios .

2.1.1 VANET-Based Systems

These systems have introduced automations in traffic monitoring in the method proposed by Elsagheer Mohamed (2019) VANET-Based Systems and automated recording and reporting system was developed to report traffic violations using GPS and GSM .As the system seen efficient in controlled environments and substantial requirement of the infrastructure is required for the deployment .

2.2 Object detection

The study Garcia-Garcia et al. (2020) explore background subtraction techniques utilized in real-life computer vision applications. It demonstrates the conflict between the impressively complex mathematical and machine learning schemes introduced in papers and their diffusion in real-life applications like traffic monitoring. Based on the shortcomings mentioned above, the authors suggest using an exhaustive survey in order to assess real-life difficulties of the system, such as challenges with different types of cameras (CCD, omnidirectional etc.), foreground objects of various kind, as well as versatile environment. They dealt with the approach that involves studying the background models that are currently being used in practical applications and comparing them with new models as to robustness and computational efficiency and memory size. Recognizing the shortcomings of the existing large-scale datasets, which do not cover all the issues arising in practical applications, the study contributes to identifying the applicability of various models. The study findings indicate the absence of sufficiently representative datasets in practice and indicate that linking theoretical models with realistic conditions could help the field progress. But one major drawback is that most of the suggested models may only be tested with simulated data and cannot be easily applied to different conditionings immediately.

The work Kalsotra and Arora (2022) offers an analytical survey of the prior and current developments in background subtraction and the difficulties to build a universal model of moving object detection for real time systems. The authors stress that background subtraction is one of the key prerequisites for achieving effective higher-level video analysis and describe the advances that deep learning, especially deep neural networks, have brought to the subject, solving significant issues. They also emphasize on the advantage of combining several features to improve the standard approaches. The study provides a brief description of the background subtraction process, various issues and articulates the availability of benchmark video databases. The performances of the state-of-the-art algorithms are discussed and recent methods analyzed to explain the reported achievements. Then the paper points out the deficiencies of the current developments like in generalization and scalability of the existing methods used in clinical view extraction and finally presents future work prospects like improving the dataset, and combining deep learning with conventional methods. Again, we can find certain drawbacks of the review: still, the authors use only available datasets and neglect some essential real-life aspects, and the proposed recommendations have not been evaluated based on experiments.

The work Zulunov et al. (2024) explores the mathematical background and the practical application approaches for the YOLO (You Only Look Once) object detection algorithm, considering its applicability for moving object detection. Components discussed include the bounding box representation, the IoU computations, MSE for representing objectness score, a post-processing method, NMS, and learning rate issues. It also discusses some highly technical concepts such as the anchor boxes, back prop, and data augmentation to enhance the prowess of YOLO's accuracy as well as versatility during dynamic scene filming from a discriminating between frames prowess. The study makes use of the best performing Python-based YOLO library implementations that provide efficient program codes. Moreover, it also embeds two algorithms including the edge detection one along with the background subtraction one for identifying the mobile objects. The results show that YOLO delivers high-real-time detection rates across multiple domains. Nevertheless, some weaknesses and drawbacks are found in this study: it is weak when dealing with special cases like high dynamic scenes or occlusion and lacks comparison of experimental results with other methods, which may lead to incomplete assessment of the proposed approaches.

The work Xu et al. (2020) puts forward a new moving object detection scheme based on sample to make background subtraction to tackle with issues like illumination variation, static foregrounds and dynamic backgrounds. A simplified motion region estimation starts with an enhanced frame difference method with block split image partition and multi-scale region based method. This method enables one to counter global illumination variations hence cancel dynamic background disturbance. Finally, the algorithm combines an improved ViBe technique with distance thresholds and time sub-sampling factor as parameters for each pixel to provide improved object tracking. This refinement is to enable the retention of static foregrounds and efficient detection of objects. Experiments were done on CDnet 2014 and Wallflower datasets.It resulted in achieving an F-measure of 0.7625.However, the algorithm may behave suboptimal when it is dealing with complex and densely populated scenes where maybe further fine-tuning of the algorithm with respect to specific outliers might be necessary.

2.3 Deep Learning

There are lots of studies which have been done on deep learning based approaches. There is a study which is given by Maity et al. (2021) has suggested a system level study for the important aspect of vehicle detection and tracking that has particularly addressed two of the major network architectures Faster R-CNN and YOLO. The purpose of this work is to review our current method classification by backbone structures and characterize relations between them, as well as reveal temporal progressions. In particular, the review includes an evaluation of the Faster R-CNN and YOLO structures as well as the proposed variations to enhance the comprehension of their functionality in the application context of ITS. The problem focused by the paper is to detect the vehicles and then track them under different conditions including low illumination and light conditions which are important in reducing the number of accidents and efficient traffic surveillance. The results outlined below show the limits of the existing approaches and indicate some avenues for future research, for instance, a method of accounting for demanding weather situations or enhancing promptness. However, the study has a restriction as most part of the research is collected from existing sources and there are likely no fresh practical implementation, and no comparative experimental evidences may be attached to support the conclusions and findings completely. This work maps out a way forward in order to come up with improvements of the vehicle detection systems.

The study by Dodia and Kumar (2023) compares performance of three different versions of YOLO v3,v5 and v7.The datasets used for experiment is open source traffic video footage. The study concludes that the YOLOv7 outperforms other model with mean Average Precision of 95.74% even though the v5 model give a balance between the speed and accuracy. The study suggests YOLOv7 model is ideal for real-time detection and its potential for further applications and integration to traffic monitoring.

Zhang et al. (2022) presents an enhanced vehicle detection system using YOLOv5 to address challenges like occlusions and small object identification .They introduced Flip-Mosaic data augmentation method to improve detection accuracy.The custom dataset from highway surveillance footages with annotations are used .The proposed method shown improvements in mean Average Precision (mAP) especially in identification of SUVs and sedans. However, possible drawbacks may include difficulties in applying the model on unseen traffic scenes, or processing specific highly complex occlusion cases. Nevertheless, the approach offers several improvements in vehicle detection for various settings.

Benjumea et al. (2021) have developed a specific lightweight vehicle detection algorithm, namely YOLOv5n-L, intending to mitigate the shortcomings of complex structures, high hardware demand constraints, and restrictions to portable devices of various existing algorithms. The proposed plan uses the depthwise separable convolution and C3Ghost to remove model parameters and increase detection speed. Further, to enhance the accuracy of the backbone network and filter out interference from the environment, a Squeeze-and-Excitation attention mechanism is introduced into the network architecture Further, for the multi-scale feature overcombing, a bidirectional feature pyramid network is adopted to fuse the features comprehensively. Performance evaluation indicates that the proposed algorithm, apart from minimizing the hard disk storage to 2,3 MB by eliminating the entire redundant weight, increases the mAP@0.5 by 1.7 percent and increases the detection rate per frame by seven percent, thus making the algorithm offer real time performance with 80 FPS. However, it still has certain drawbacks: it may be difficult to positively affect complex detection tasks, or attain similar effectiveness when working with increased volumes of data. The method presents an interesting solution for effective vehicle detection for lightweight and efficiency especially in the mobile and low resource environments.

2.4 Wrong way detection

Usmankhujaev et al. (2020) has proposed a system identifying violators of the wrong way direction uses video imaging together with deep learning methods for detection, tracking, and validation. For the detection of vehicles we have used the YOLOv3 deep learning model with the dataset collected from twelve videos, filmed at different times of the day, using the split between the training and testing set 70/30. The tracking is done using Kalman filtering, which is an estimator and an "entry-exit" algorithm to verify the tracks and test wrong-way driving. There was a considerable problem in the context of this system concerning reliable identification and subsequent validation of the corresponding vehicle movements under different lighting and environmental conditions. Nevertheless, the stated troubles limited wrong-way vehicle identification accuracy to 91.98 percent. There was also a specific subset containing only such data that corresponds to cars moving in the wrong direction that allows for effective validation of detection performance. The dataset which has been used in this study has taken from a fixed CCTV camera.

Rahman et al. (2020) proposed a real time wrong way detection using the YOLOv3 and also centroid tracking is used for identification of vehicle movements. The system used pretrained model trained using COCO dataset .The system identifies violation in specified region and identifies the violations in the specified region only. Even though the system shows near perfect accuracy in their test conditions.The study only considers single version of YOLO and custom training of datasets are not employed to enhance detection in various conditions .

Suttiponpisarn et al. (2022) proposes wrong way driving detection using deep learning and image processing techniques. It combines Road lane boundary detection from CCTV(RLB-CCTV) algorithm for identifying road boundaries Majority based Correct Direction Detection to detect incorrect driving direction. YOLOv4 Tiny was used Fast-MOT is used for tracking. The pre-trained model of YOLOv4-Tiny on standard COCO dataset is used for vehicle detection. The system is tested primarily on motorcycles violations .The method obtain an accuracy of 96.61% but its limitation include good lighting, require straight roads.

2.5 Research Gaps

The existing works in wrong way detection uses RCNN and previous versions of YOLO also these works are done with standard datasets like COCO which may not include vehicle data from real traffic conditions. The custom dataset increase the data diversity .The data augmentation are not seen done on most of the works which can simulate different environmental conditions including poor lighting and bad weather conditions. The gap suggests the requirement of research using custom dataset ,data augmentation and other later versions of YOLO.

3 Research Methodology

The goal of this study will be to design a real time system for identifying wrong way movements of the vehicles on the road and thus help in reducing fatal incidents. Wrong-way driving is a major safety problem, which could cause occurrences with fatal outcomes, and hence needs proper detection to help in traffic management. In this study, YOLOv7 is used as an object detection algorithm with the primary interest being on vehicle detection, their motion, and infraction of established traffic flow directions. Performance indicators include the correctness achieved by the system, its real time capability and functional capacity under different environmental conditions. The configuration of the work corresponds to the expectations of its goal as a smart traffic monitoring system and the improvement of road safety.

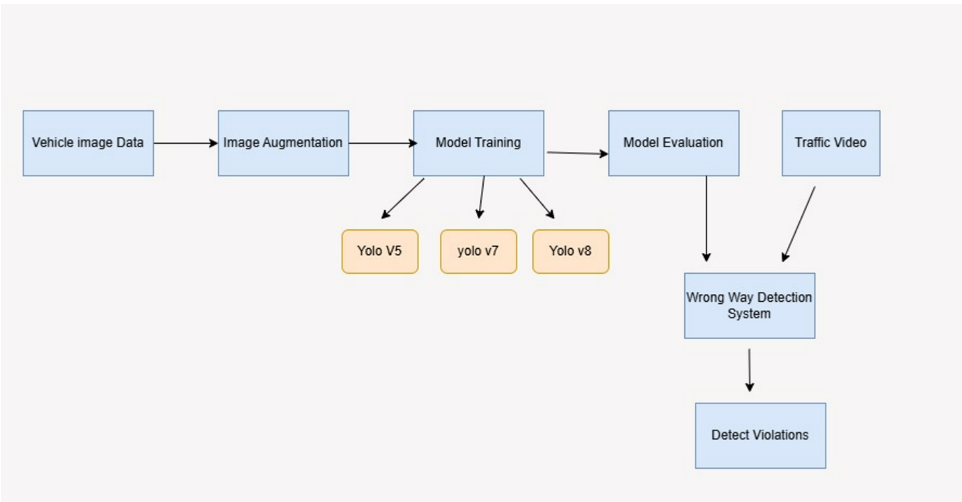

Figure 1: Workflow diagram of proposed system

3.1 Dataset Description

The dataset employed in this vehicle detection study is obtained from Roboflow [1] containing 1961 images and labels with six classes of vehicle. Such classes include different sorts of automobiles so that it is possible to have a wide range of training and evaluation sets.The annotations file are saved in different folder with same label as image file

[1]Dataset url: `https://universe.roboflow.com/aliyahhalim/vehicle-detection-q8q4n`

name and is in '.txt' file format. The dataset which is downloaded is already split into train,validation and test sets which is suitable for YOLO.The data is downloaded as test ,validation and train .The train set have 1355 images while validation contains 412 images and test set contains 194 images

Figure 2: snapshot of images in dataset

3.2 Data Augmentation

Data augmentation was conducted to increase the range of the datasets and to introduce certain conditions inside the framework of the real-life situation to make the model more resistant. To augment the original images and their associated bounding boxes, a sequence of transformations was applied using the imgaug library. Further, to simulate different lighting conditions per-channel gamma adjustments were applied by adding random multiplicative factors which range between 0.8 and 1.2. To mimic low to high contrast conditions, adjustments to image contrast ranging from 0.75 to 1.5 was applied. These changes were in hue and saturation and were standard at between -20 and 20, as previously noted, simulating conditions of color shift resulting from external influencers. Geometric transformations were scaling which allowed random scaling factors of between 0.8 and 1.2, translation (-20% to 20%), rotation (-30° to 30°) and shear (-10° to 10°) thus increasing the number of objects in different orientations within the dataset. Two types of blur were added to the images; Gaussian and motion blur to accurately replicate scenes that may be out of focus or scenes that contain moving objects such as the cars. Further, JPEG compression with inter Picture control with the compression ratio of 70 –99% portrayed low quality images common in real life.

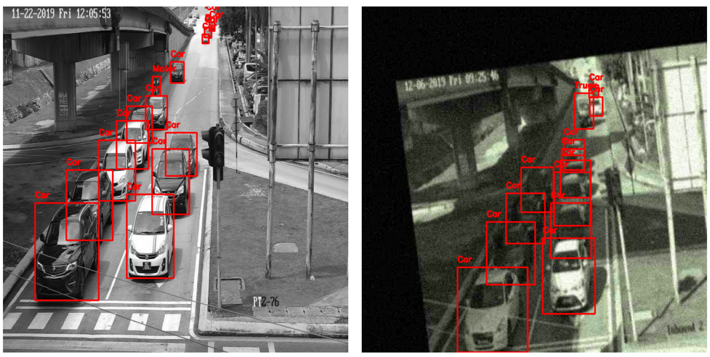

Figure 3: Image before and after augmentation

3.3 Model Building

For the purpose of object detection the underlying base model is YOLOv7 We fine-tune this model to accurately identify and classify vehicles using the presented dataset. Learning rate, momentum and weight decay factors are well adjusted so as to make performance better. Some of the parameters in the convolutional layer are modified in order to enhance the extraction of features. We compared YOLOv5, YOLOv7, and YOLOv8, and YOLOv7 proves to be the most efficient in detectors per second with high accuracy. In this step high performance GPU enabled resource is used for training the processed dataset as the model training requires high performance. Different Yolo versions like Yolov5, Yolov7 and Yolov8 models are used for training the dataset. The dataset is trained with pretrained weights of respective models, and the result of each model are evaluated and out of which the yolov7 results are seen better compared to other versions. The custom trained model is saved for detecting vehicle in the detection system.

3.4 Model Evaluation

Performance features are tested using mean Average Precision (mAP@0.50) which is the average precision calculated across all classes at IOU threshold of 0.5 precision and recall and normalized confusion matrix is plotted and evaluated.

The Average Precision (AP) is given by:

$$AP = \int_0^1 P(r)\, dr$$

The Mean Average Precision at IoU = 0.5 (mAP@0.5) is given by:

$$mAP@0.5 = \frac{1}{N_c} \sum_{i=1}^{N_c} AP_i$$

Where:

- $P(r)$ is the precision as a function of recall r,
- N_c is the total number of classes,
- AP_i is the Average Precision for class i.

Precision is the ratio of correctly predicted positive prediction out of all predicted positive observation (Sammut and Webb; 2011). Recall which is the ratio of orrectly predicted positive prediction out of all actual positive observation (Sammut and Webb; 2011).

3.5 Using the Model for building Wrong Way detection System

In this phase the model which is trained and evaluated is used for building the vehicle detection system. The model is saved in 'pt' format is loaded into the python application and is used for classification of vehicles from input video .The process involved is discussed below.

3.5.1 Initialize Video Capture

The user inputs the video file path after which video capture is initiated using OpenCV. OpenCV was used in reading frames in the video and identifying and tracking them in a sequential manner. To detect the wrong way vehicles as they move, real-time video processing is critical. This setup guarantees that the video files can be in any format so that they can be processed under the YOLOv7 model hence the constant processing of frames for vehicle detection. Video capture initialization is the key to frame by frame analysis of the video stream.

3.5.2 Define Reference Direction

In this section, the user is simply invited to specify time instants within the video and to point at the allowed direction for vehicles. We then compute the reference direction using OpenCV's mouse callback functionality by finding the vector AB and normalising it. This vector shows where vehicles are allowed to move in a specific environment. By defining which direction is the reference in the sequence, then further comparing car movement with regard to this vector, the model allows for the identification of vehicles moving in the opposite direction. This step is critical, in ensuring higher accuracy in tracking the violations.

3.5.3 Process Video Frames and Detect Violations

To normalize the input for detection the system resizes the image to the specified width and height for every frame in a video. YOLOv7 applies inference to each frame detecting vehicles. The detected vehicles are painted in green boxes.The detected vehicles are identified using IDs, which makes it easy for continuity when linking it to subsequent frames.Then centroid for the vehicle is calculated. This centroid information is critical for monitoring vehicle movement. The movement of the vehicle is calculated based on the displacement between its current and previous centroids making a movement vector.This movement vector is compared with the reference direction given by the user thus the system identifies violations and violated vehicles are painted in red bounding boxes.Thus identifies the wrong way driving.

4 Design Specification

In the Model Training phase horizontally at the same time, they use three versions of YOLO, which include YOLO V5, YOLO V7 & YOLO V8. Model Evaluation identifies that YOLOv7 is the best performer based on the mAP and precision/recall assessment. It further moves to Detect Violations whereby the trained model implementing real-time with OpenCV is realized to give the overall wrong-way detection system capable of monitoring and alerting traffic violations effectively.

4.1 Model Architecture of YOLO Models

The version used for YOLOv5 is the small version.The architecture for the model is shown in figure 4 .The model contains backbone ,Neck and Head. Key features from input data is extracted by backbone. The feature fusion enhancement is done by the Neck. Finally, the head component is responsible for making final predictions which includes bounding boxes, object class probabilities and confidence scores.(Ultralytics; 2024)

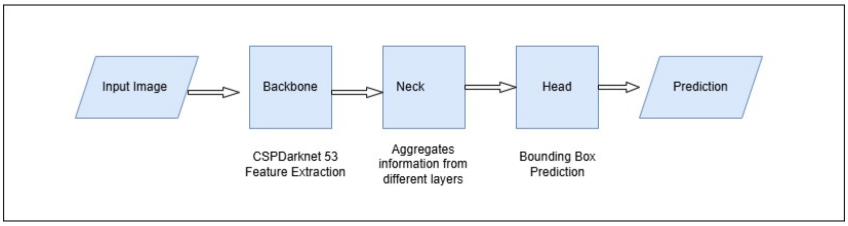

Figure 4: Architecture for YOLOv5

The YOLOv7 is the improved version of YOLOv5 .The backbone contains Extended Efficient Layer Aggregation Network also the Neck combine FPN and PANet for feature aggregation. The head have a decoupled design for better performance. This version incorporates reparameterization to merge layers .The architecture of the model is shown in figure 5.(Wang et al.; 2022)

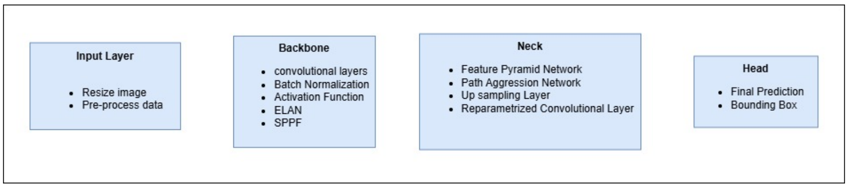

Figure 5: Architecture of YOLOv7

The YOLOv8 version has a Cross stage Partial Network(CSPNet) backbone and C2f models for efficient feature extraction. The neck have a combination of FPN and PANet for feature aggregation. Also the detection head is anchor free. The architecture of the model is given in figure 6.

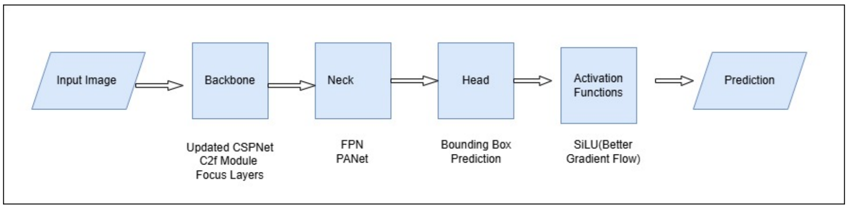

Figure 6: Architecture of YOLOv8

5 Implementation

The image dataset after being downloaded into the training environment containing different vehicle classes undergone data augmentation are then further trained for building vehicle detection model .Three different YOLO models are trained and the trained model is further saved and each versions are evaluated using evaluation parameters and the best trained model among is used in the violation detection application .

5.1 YOLOv5 Model

Initially YOLOv5 version is used for training of processed dataset. The model is imported into the training environment from the original git repository. Also pre trained weight "yolov5.pt". Input image size is standardized to 416*416 for consistency .The batch size is set to 16 to optimise GPU usage and to enhance training efficiency .The learning rate which will be dynamically update is initially set to 0.01.The yolov5 model uses Stochastic Gradient Descent optimiser with momentum 0.937 and weight decay 0.0005. The model is trained for 100 Epochs feeding batches of labelled images for training .

5.2 YOLOv7 Model

The YOLOv7 model is downloaded and weights for transfer learning are downloaded from orginal git repository.The dataset image size is set to 416*416 .The model is trained for 100 epochs .The initial learning rate is set to 0.01 and adjust dynamically during training to optimize convergence.The batch size is set to 16 and the optimizer used is Stochastic Gradient Descent having momentum 0.937 and weight decay 0.0005.

5.3 YOLOv8 Model

The YOLOv8 model is accessed from ultralytics library .The augmented dataset is trained using the YOLOv8s pre-trained weight .The dataset for training is specified through data.yml file.The training configuration includes 100 epochs.All images are resized to 416*416 which enables computational efficiency.The initial learning rate is set to 0.01 and adjust dynamically during training to optimize convergence.The batch size is set to 16 and the optimizer used is AdamW having learning rate 0.001 ,momentum 0.9 and weight decay 0.0005.

5.4 Python Application for Detection

The application process video file processing allowing the user to enter the allowed direction of vehicle movement. The vehicles detected using yolov7 model are tracked across frames using their centroids and their movement direction is calculated and if the movement is against reference direction, it is flagged as violation and are shown in red boxes and other allowed direction movements are shown in green boxes. The system uses OpenCV for video processing and user interaction. The vehicle tracking is managed by assigning unique IDs and violations are flagged based on movement vector relative to reference direction. The system also displays real time violation count and save image of violation in a specified folder.The python application workflow and the GUI of the application are shown in Figure 7 and Figure 8 respectively

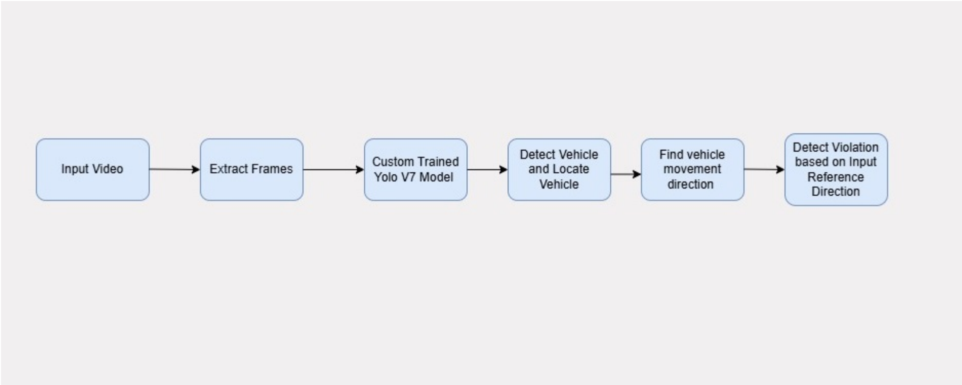

Figure 7: Python Application Design

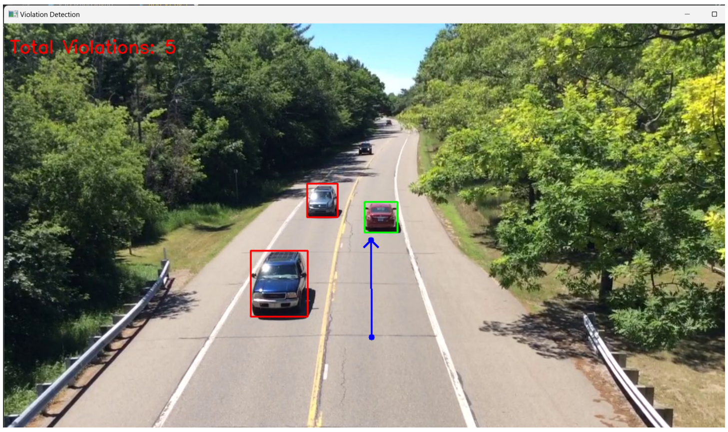

Figure 8: Graphical user interface of Detection system

5.5 Tools and Technologies

The project uses advanced tools for model training and application deployment

1. **Google Colab with T4 GPU:** A cloud-based environment used for model training with NVIDIA T4 GPU, which enables resource-intensive model training of image data.

2. **PyTorch:** PyTorch is used to load a custom-trained YOLOv7 model into the Python application.

3. **OpenCV:** It is used for video processing and user interface. It enables frame resizing, object tracking, and drawing bounding boxes. Additionally, OpenCV enables interactive user input for defining the reference direction for violation detection.

4. **Python:** Python is used as the primary programming language for the project. Its extensive library support and integration capabilities ensure the implementation of different tasks, including model training and application development.

This combination of tools and technologies ensues efficient pipeline from training object detection model to deploying functional application for vehicle tracking and violation detection.

6 Evaluation

The comparison study of the training results of YOLO models are done in the section.

6.1 Case Study 1: Yolo-V5 Results

Recall-Confidence(Figure9) is demonstrated graphically, being an analysis of various vehicle class recall using a machine learning model. This graph indicates the recall on the Y axis against the confidence of each class such as Ambulance, Bus, Car, Motor, Truck, and Van on the X axis. Hence the major observation is that the "all classes 0.87 at 0.000" curve has the highest recall confidence tradeoff, which shows that the model has high predictive precision with any of the class types of vehicles. The figure 9 also represents the precision-recall curve of a classification model with respect to vehicle classes. The blue curve with label "all classes 0.793 mAP@0.5" corresponds to the total performance with the mean average precision of 0.793 on a per-class bases at intersection-over-union threshold of 0.5.

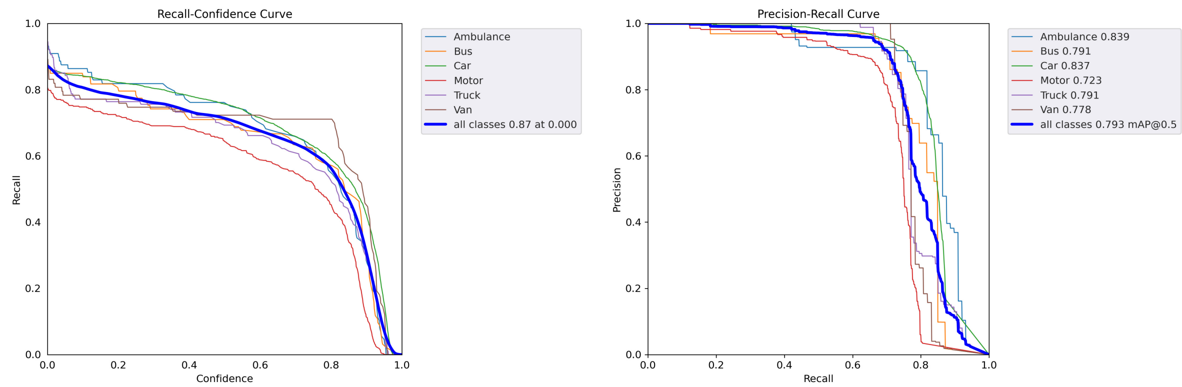

Figure 9: Recall confidence and Precision-Recall Curve YOLOv5 model

The precision confidence curve(Figure 10)of the Yolov5 object detection model for many vehicle classes. The blue curve denotes the performance of the whole model where precision of the model is 1.00 at the confidence threshold of 0.971 for all classes. The figure 10 also presents F1-Confidence curve in the yolo v5 object detection model by different car categories. The last curve is the aggregate of all classes with an F1 measurement of 0.80 for the model at a confidence of 0.484.

The confusion matrix(Figure 11) shows models performance across different classes.The normalized confusion matrix is plotted . Classes like Ambulance and car shows accuracy of 82% and 82% correct classifications , while others like truck and bus face misclassification challenges.Even though strong performance the car class have a confusion with motorcycle class and background. The model has notable confusion between similar classes such as motor being misclassified as car. Also Van and truck showing overlap .The results suggest model performs well for some classes.

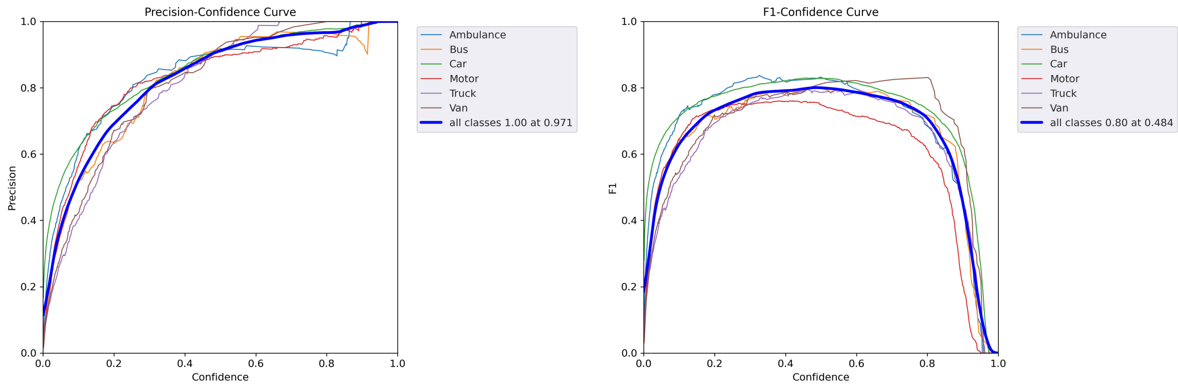

Figure 10: Precision-confidence and F1-Confidence Curve

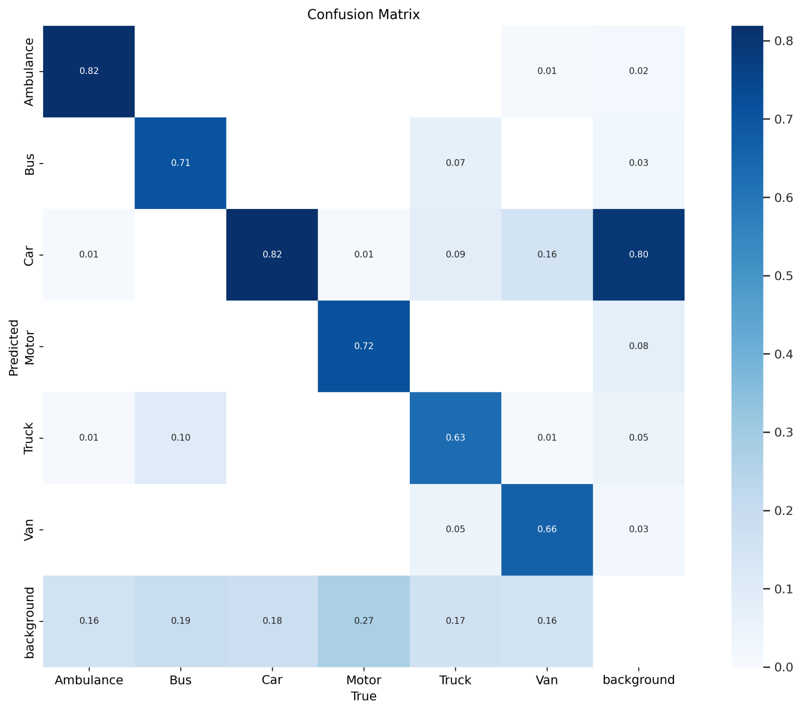

Figure 11: Normalized Confusion Matrix(YOLOv5)

The YOLOv5s has 182 layers, and 7,260,003 parameters that allow for fast vehicle detection while gradients computation. The dataset used for testing contains 412 images with 3,952 instances of six vehicle classes. The model got the mAP@50 of 0.793 meaning the model presented high detection rates. For the individual classes, the highest mAP@0.50 values were achieved with the Ambulance class at 83.9%, and the Car class at 83.7%. Accuracy and recall point towards good detection, especially for the most frequent class Car and class Motor, while the performance is less satisfactory for class Truck and Van because of their less occurrence and the variations involved. The performance metric for the model for different classes is shown in Table 1

Table 1: Performance Metrics for YOLOv5 Model

Class	**Instances**	**Precision (P)**	**Recall (R)**	**mAP@50**
All Classes	3952	0.913	0.715	0.793
Ambulance	88	0.918	0.761	0.839
Bus	93	0.917	0.709	0.791
Car	3121	0.914	0.757	0.837
Motor	440	0.896	0.645	0.723
Truck	127	0.907	0.693	0.791
Van	83	0.924	0.723	0.778

6.2 Case Study 2: Yolo-V7 Results

The results of the YOLOv7 model training are discussed The Precision-Recall Curve (Figure 12) in a Yolo v7 object detection model for all classes mAP@0.5 is 0.876 represents the ability of the model and has an average precision of 0.876 at an intersection overlap of 0.5. From recall confidence curve (Figure 12) shows the value as 0.94 at 0.00 which indicate models sensitivity across confidence thresholds. The classes Ambulance ,car and van maintains high recall even at lower confidence thresholds. From the precision-confidence curve (Figure 13 give the value “1 at ”0.962” classes Ambulance, Car and Van maintains high precision across range of thresholds while bus and motor exhibits fluctuations.

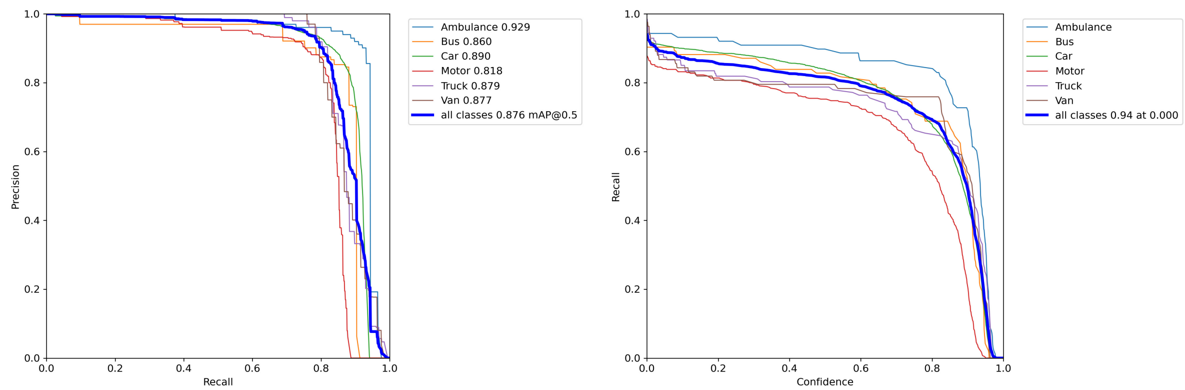

Figure 12: Precision-Recall and Recall-Confidence Curve(YOLOv7)

From the normalized confusion matrix (Figure 14) the performance of the yolov7 object detection model for different vehicle type is depicted. The higher diagonal elements show that most of the vehicles are classified correctly, and the off diagonal elements shows how wrong or how many times a class has been misclassified by the classifier. It also details out a matrix of how well the model was able to identify the different types of vehicles correctly. The confusion matrix shows models strong performance for classes “Ambulance”,”Car” and “Van”also having minor mis-classification issue with other classes.

The vehicle detection dataset is assessed with an outstanding performance of the YOLOv7 model under 412 images containing 3,952 instances of seven classes of vehicles. It obtained an mAP@0.50 of 0.876, which proved the strong ability of the network at detecting instances of VS. This can be seen with the results giving the Precision Recall

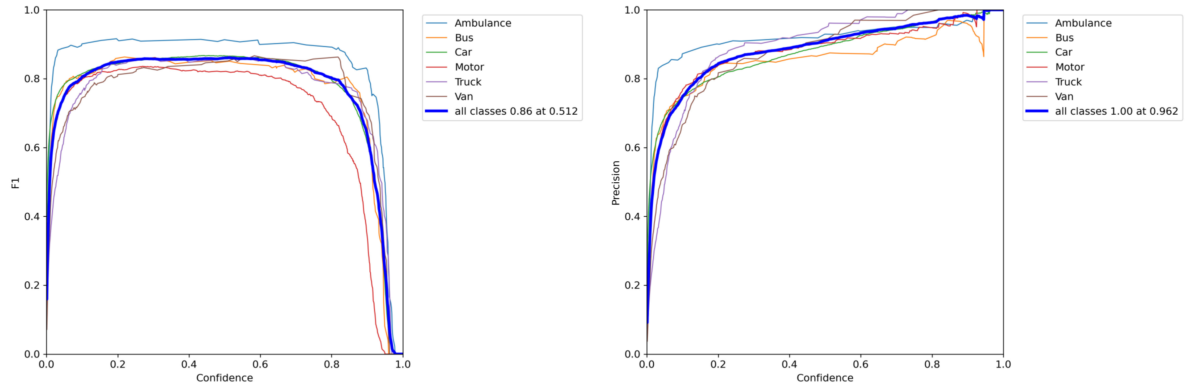

Figure 13: F1-confidence and Precision-Confidence Curve(YOLOv7)

values signifying the exactness and recall the model has regarding the identification of true positives, with the Ambulance class giving the highest mAP@0.50 in the test set of 0.929. Among all the models it turns out that this one is the best in regard to its mAP@0.50 , which proves its somewhat higher ability to detect and describe the type of vehicle.The results are tabulated in Table 2

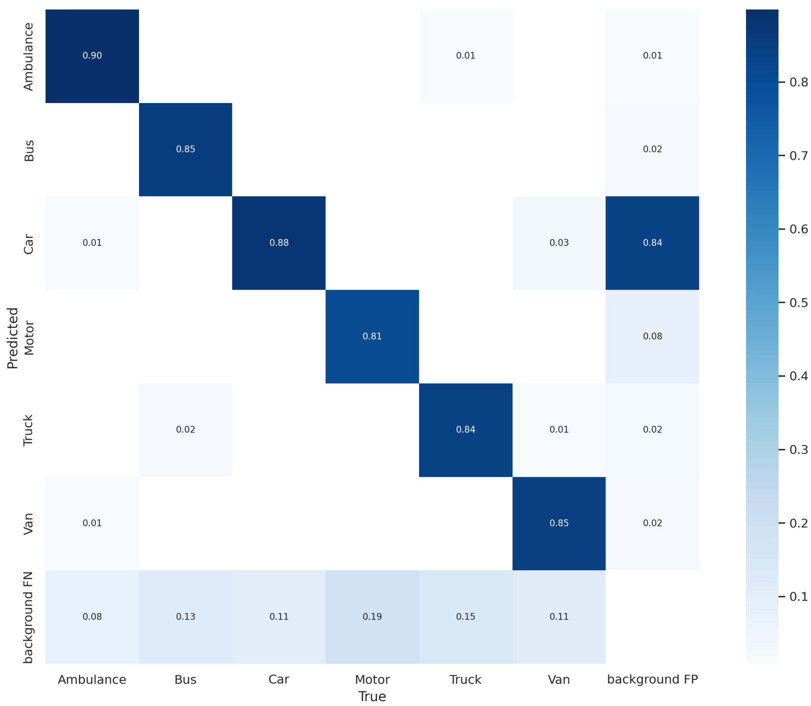

Figure 14: Normalized Confusion Matrix (YOLOv7)

Table 2: Performance Metrics for YOLOv7 Model

Class	**Instances**	**Precision (P)**	**Recall (R)**	**mAP@50**
All Classes	3952	0.914	0.815	0.876
Ambulance	88	0.929	0.894	0.929
Bus	93	0.875	0.828	0.860
Car	3121	0.898	0.835	0.890
Motor	440	0.904	0.750	0.818
Truck	127	0.962	0.787	0.879
Van	83	0.919	0.795	0.877

6.3 Case Study 3: Yolo-V8 Results

The mAP @0.5 for all classes is 0.866.From the recall confidence curve (Figure 15 we can understand that the overall recall peaks at 0.92 at a confidence of 0.0.Some classes have strong recall. From precision confidence curve it can be seen that model achieves peak precision of 1 at a confidence threshold of 0.987.Also the F1 score is seen high at 0.85 at a confidence threshold of 0.459.Also the confusion matrix for the model is shown in figure. Some classes are showing strong performance,also slight miss-classifications are seen for other classes.

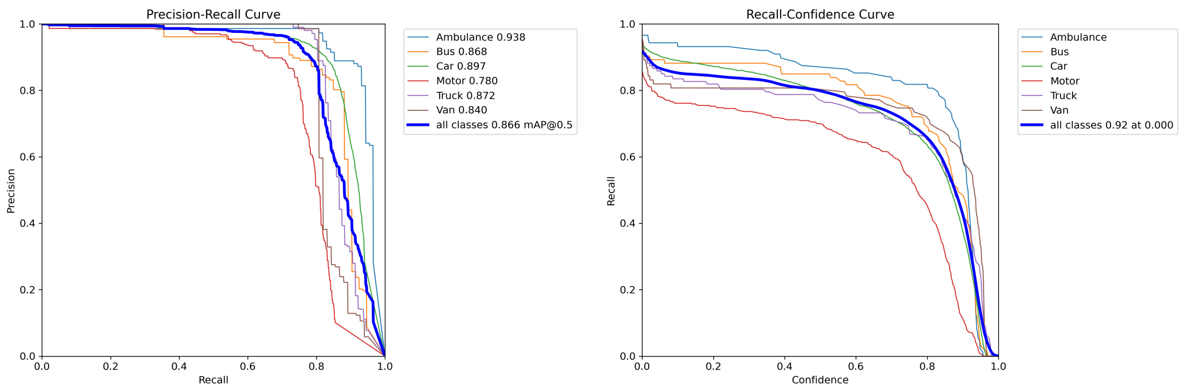

Figure 15: Precision-Recall Curve and Recall-Confidence Curve

Results of the experiments conducted for the YOLOv8 model revealed that an overall mAP@0.50 of 0.856 was obtained on a vehicle detection dataset. The model has 168 layers of depth and 3 006 818 parameters. As measures of detection accuracy, precision and recall are used, and the best value of mAP@50 is achieved by the class “Ambulance”, 0.929. Table 3 shows the performance metrics of the YOLOv8 model.

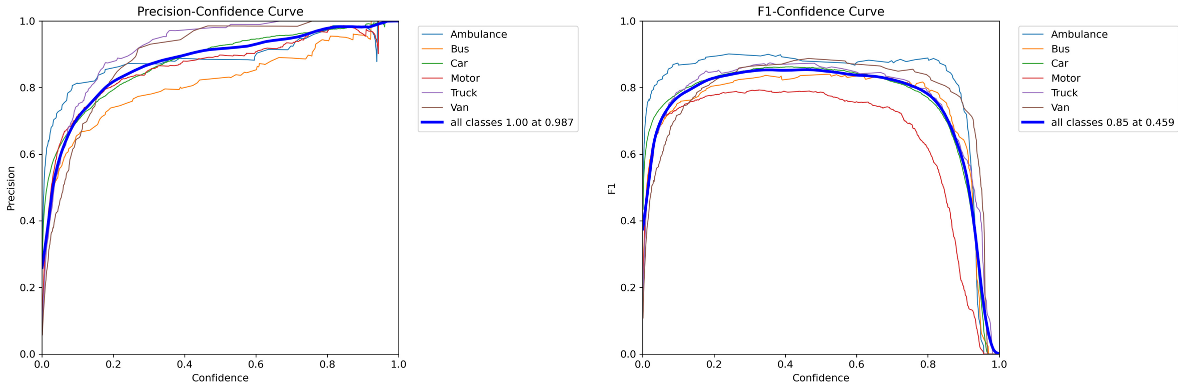

Figure 16: Precision-confidence Curve and F1-Confidence Curve

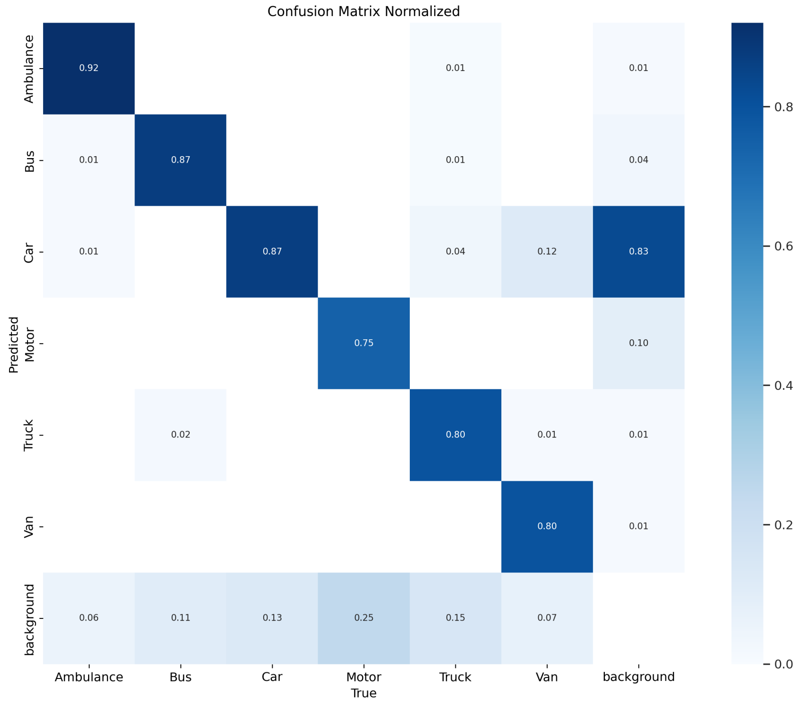

Figure 17: Confusion Matrix YOLOv8

Table 3: Performance Metrics for YOLOv8 Model

Class	**Instances**	**Precision (P)**	**Recall (R)**	**mAP@50**
All Classes	3952	0.911	0.807	0.866
Ambulance	88	0.885	0.875	0.938
Bus	93	0.824	0.849	0.868
Car	3121	0.913	0.814	0.897
Motor	440	0.887	0.709	0.780
Truck	127	0.978	0.787	0.872
Van	83	0.980	0.807	0.840

6.4 Discussion

While comparing the three models based on performance parameter for the specific dataset chosen for model building YOLOv7 model out performs other two models even though the performance of YOLOv8 model is almost equal the mAP@0.50 value for the former is 0.876 as compared to the value 0.866 for YOLOv8.Also the precision and Recall values for the YOLOv7 model seen better compared to the other two.Further consistent strong performance for are shown for key classes .Also fewer background misclassifications are shown by the YOLOv7 model which helps in reducing false positive rate in cluttered environments.Also computational efficiency and faster inference time makes it more suitable for real world deployments. Although YOLOV8 have advanced features and slight improvements in minority class detection,the higher overall performance makes it more effective model for this dataset.

6.5 Evaluation of Vehicle Detection System

The python application developed with the best model ie YOLOv7 for detecting wrong way violations.The system is tested using two sample video traffic footages where vehiches flowing in both directions.The vehicles moving towards north are assumed to be correct direction and moving opposite direction is taken as violation .The table below summarizes the results,comparing actual violations detected by the system with expected violations based on our assumptions.The figures are arrived by manually analyzing the video and the detection flagged .From results tabulated in Table 4 for video 1 the system is able to attain an accuracy of 100% while for the second there is a false positive value so accuracy dipped to 88.9% .So Taking the average based on the two videos the system have an accuracy of 94.5% which is the average of both and this figure is specific to the video input used.

Video	**Actual Violations**	**Detected Violations**	**False Positive**
Video 1	6	6	0
Video 2	9	8	1

Table 4: Comparison of actual and detected violations

7 Conclusion and Future Work

Lastly, this work proposes an efficient method to identify wrong way vehicles utilizing YOLOv7 that employs real-time applications of detect and track of abnormal traffic behaviour in video streams. Thus, the described environment setup provides all the necessary libraries and chooses a suitable device to predict vehicles on the detected video frames. It is important that with reference direction system defined through the user will be a good way of measuring vehicle's movement so that violation may easily be detected. Overall performance of the system is demonstrated in its applicability with multiple classes of vehicles and significant enhancements in detection and tracking through multiple conditions.The bounding boxes on the visualization make it possible to have real-time analysis of the results as well as recorded violation history. This methodology also shows how future work can combine deep learning models such as the YOLOv7 with video analysis for application in traffic monitoring. This makes it effective to be used in improving the traffic control, increasing measures of safety, and backing up of regulations. In general, the given paper effectively showcases the possibility of using modern computer techniques in solving the problem of modern traffic management.

7.1 Future Work

However, there are some drawbacks in the present research that need to be discussed in further studies: A significant limitation of the work is the use of a single dataset, which may not contain sufficient variation in traffic conditions and vehicle models.Also system is only tested with two sample videos and not tested in more complex environments. For future work, more and varied samples encompassing different environments, varying light conditions, and greater range of traffic scenarios could be used to further fine tune the generalization of the model. Furthermore, despite the effectiveness of YOLOv7, expanding our study to other YOLO versions and other better models and architectures including EfficientDet and Transformer-based models might allow for improved detection performance and more rapid inference times. One of the problems is that the definition of the reference direction is made by selecting points by hand, which could be done automatically for better performance in the dynamic context. That is why it is possible to increase the precision of tracking and detection with the use of such algorithms as Kalman Filters or DeepSORT for deeper vehicle tracking. Moreover, one can think about the possibility of improving this ability of the model and perform it in real-time to accomplish the task of identifying wrong-way vehicles by studying and experimenting with the different ways of using hardware acceleration features and employing other edge computing devices to accelerate the process of processing in an application used for real-time purposes. Lastly, in future studies one could include data of other types, for instance radar or LIDAR, to improve the detection robustness especially during the night or in rainy weather.

References

Akib, I., Khan, M., Islam, A., Sharif, N. and Hasan, M. (2023). *Integrated Traffic Violations Detection System for the Highways of Bangladesh*, PhD thesis, Department of Electrical and Electronics Engineering (EEE), Islamic University of Technology (IUT), Board Bazar, Gazipur-1704, Bangladesh.

Benjumea, A., Teeti, I., Cuzzolin, F. and Bradley, A. (2021). Yolo-z: Improving small object detection in yolov5 for autonomous vehicles, *arXiv preprint arXiv:2112.11798* .

Dodia, A. and Kumar, S. (2023). A comparison of yolo based vehicle detection algorithms, *2023 International Conference on Artificial Intelligence and Applications (ICAIA) Alliance Technology Conference (ATCON-1)*, pp. 1–6.

El Atigh, H. and Özer, Z. (2021). Traffic violation detection systems : A survey, *Traffic violation detection systems : A survey.*

Elsagheer Mohamed, S. A. (2019). Automatic traffic violation recording and reporting system to limit traffic accidents : Based on vehicular ad-hoc networks (vanet), *2019 International Conference on Innovative Trends in Computer Engineering (ITCE)*, pp. 254–259.

Garcia-Garcia, B., Bouwmans, T. and Silva, A. (2020). Background subtraction in real applications: Challenges, current models and future directions, *Computer Science Review* **35**: 100204.

Kalsotra, R. and Arora, S. (2022). Background subtraction for moving object detection: explorations of recent developments and challenges, *The Visual Computer* **38**(12): 4151–4178.

Li, L., Lin, Y., Du, B., Yang, F. and Ran, B. (2020). Real-time traffic incident detection based on a hybrid deep learning model, *Transportmetrica A: Transport Science* **18**: 1–21.

Maity, M., Banerjee, S. and Chaudhuri, S. (2021). Faster r-cnn and yolo based vehicle detection: A survey, *2021 5th International Conference on Computing Methodologies and Communication (ICCMC)*, IEEE, pp. 1442–1447.

Rahman, Z., Ami, A. M. and Ullah, M. A. (2020). A real-time wrong-way vehicle detection based on yolo and centroid tracking, *2020 IEEE Region 10 Symposium (TENSYMP)*, pp. 916–920.

Reddy, A., Arif, S. and Varma, J. (2024). Wrong way vehicle detection using deep learning techniques.

Redmon, J., Divvala, S., Girshick, R. and Farhadi, A. (2016). You only look once: Unified, real-time object detection, *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 779–788.

Ren, S., He, K., Girshick, R. and Sun, J. (2015). Faster R-CNN: Towards real-time object detection with region proposal networks, *Advances in Neural Information Processing Systems (NIPS)*, pp. 91–99.

Sammut, C. and Webb, G. I. (2011). *Encyclopedia of Machine Learning*, Springer, Boston, MA.

Suttiponpisarn, P., Charnsripinyo, C., Usanavasin, S. and Nakahara, H. (2022). An enhanced system for wrong-way driving vehicle detection with road boundary detection algorithm, *Procedia Computer Science* **204**: 164–171. International Conference on Industry Sciences and Computer Science Innovation.
URL: *https://www.sciencedirect.com/science/article/pii/S187705092200758X*

Ultralytics (2024). Yolov5 documentation. Accessed: 2024-12-11.
URL: *https://docs.ultralytics.com/yolov5/*

Usmankhujaev, S., Baydadaev, S. and Woo, K. (2020). Real-time, deep learning based wrong direction detection, *Applied Sciences* **10**(7): 2453.

Wang, C.-Y., Bochkovskiy, A. and Liao, H.-Y. M. (2022). Yolov7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors, *arXiv preprint arXiv:2207.02696* .

Xu, Y., Ji, H. and Zhang, W. (2020). Coarse-to-fine sample-based background subtraction for moving object detection, *Optik* **207**: 164195.

Zhang, Y., Guo, Z., Wu, J., Tian, Y., Tang, H. and Guo, X. (2022). Real-time vehicle detection based on improved yolo v5, *Sustainability* **14**(19): 12274.

Zulunov, R., Soliyev, B., Kayumov, A., Asraev, M., Musayev, K. and Abdurasulova, D. (2024). Detecting mobile objects with ai using edge detection and background subtraction techniques, *E3S Web of Conferences*, Vol. 508, EDP Sciences, p. 03004.