electronics

MDPI

Article

Deep Learning-Based Object Detection and Scene Perception under Bad Weather Conditions

Teena Sharma [1,*], Benoit Debaque [2], Nicolas Duclos [2], Abdellah Chehri [1,*], Bruno Kinder [2] and Paul Fortier [3]

[1] Department of Applied Sciences, University of Quebec in Chicoutimi, Saguenay, QC G7H 2B1, Canada
[2] Thales Communications and Security SAS, Quebec, QC G1P 4P5, Canada; benoit.debaque@thalesgroup.com (B.D.); nicolas.duclos-hindie@thalesgroup.com (N.D.); bruno.kinder@thalesgroup.com (B.K.)
[3] Department of Electrical and Computer Engineering, Laval University, Quebec, QC G1V 0A6, Canada; fortier@gel.ulaval.ca
* Correspondence: teena.sharma1@uqac.ca (T.S.); achehri@uqac.ca (A.C.)

Abstract: Large cities' expanding populations are causing traffic congestion. The maintenance of the city's road network necessitates ongoing monitoring, growth, and modernization. An intelligent vehicle detection solution is necessary to address road traffic concerns with the advancement of automatic cars. The identification and tracking vehicles on roads and highways are part of intelligent traffic monitoring while driving. In this paper, we have presented how You Only Look Once (YOLO) v5 model may be used to identify cars, traffic lights, and pedestrians in various weather situations, allowing for real-time identification in a typical vehicular environment. In an ordinary or autonomous environment, object detection may be affected by bad weather conditions. Bad weather may make driving dangerous in various ways, whether due to freezing roadways or the illusion of low fog. In this study, we used YOLOv5 model to recognize objects from street-level recordings for rainy and regular weather scenarios on 11 distinct classes of vehicles (car, truck, bike), pedestrians, and traffic signals (red, green, yellow). We utilized freely available Roboflow datasets to train the proposed system. Furthermore, we used real video sequences of road traffic to evaluate the proposed system's performance. The study results revealed that the suggested approach could recognize cars, trucks, and other roadside items in various circumstances with acceptable results.

Keywords: smart cities; intelligent transportation systems; object detection; YOLO

check for updates

Citation: Sharma, T.; Debaque, B.; Duclos, N.; Chehri, A.; Kinder, B.; Fortier, P. Deep Learning-Based Object Detection and Scene Perception under Bad Weather Conditions. *Electronics* **2022**, *11*, 563. https://doi.org/10.3390/electronics11040563

Academic Editors: Dongkyun Kim, Qinghe Du, Mehdi Sookhak, Lei Shu, Nurul I. Sarkar, Jemal Abawajy and Francisco Falcone

Received: 16 December 2021
Accepted: 10 February 2022
Published: 13 February 2022

1. Introduction

Many advanced artificial intelligence-based applications, such as smart autonomous or self-driving vehicles [1], smart surveillance [2], and smart cities [3], have been considered as the foundation for sustainable smart cities and societies. Object detection plays an essential role in developing smart cities in normal traffic conditions or autonomous environments. It can extract helpful and precise traffic information for traffic image analysis and traffic flow control. This information includes vehicle count, vehicle trajectory, vehicle tracking, vehicle flow, vehicle classification, traffic density, vehicle velocity, traffic lane changes, and license plate recognition [4]. Furthermore, the information can help detect other road assets such as pedestrians, vehicle types, people, traffic lights, earthworks, drainage, safety barriers, signs, lines, and the soft estate (grassland, trees, and shrubs) using different objects detectors.

Many studies and surveys presented various object recognition techniques in vehicular environments, out of which the three most common detection approaches are: manual, semi-automated, or fully automated [5]. The traditional methods for collecting information about objects present on roads involve manual and semi-automated surveys. In a manual approach, a visual inspection of the objects present on the streets/roads is done either through walking or driving along streets/roads using a slow-moving vehicle. Such inspection suffers from a subjective judgment of inspectors [6]. It requires a significant human intervention which has proven time-consuming, given the extensive length of road networks and the number of objects. Moreover, inspectors must often be physically present in the travel lane, exposing themselves to potentially hazardous conditions.

In semi-automated object detection procedures [7,8], the objects on the roads/streets are collected automatically from a fast-moving vehicle and the collected data is processed in workstations at the office. This approach improves safety but still is based on the policy, which is very time-consuming. Fully automated object detection techniques often employ vehicles equipped with high-resolution digital cameras and sensors [9]. The collected images/videos are then processed using pretrained recognition software-based models identifying vehicles and surrounding objects. The data processing may be accomplished during data collection or later as postprocessing at the office. Specialized vehicles used for automatic object detection are usually equipped with multiple sensors such as laser scanners and LiDAR cameras to capture road assets. Vehicle-based traffic detection is standard as they enable efficient and faster inspection of the objects.

Deep neural networks have considerably improved the performance of smart autonomous or self-driving cars, smart surveillance, and smart city-based applications compared to ordinary traditional machine learning-based approaches. Deep learning, based on neural networks, is a more advanced kind of machine learning that offers solutions in many complex application models using traditional statistical methods [10]. For example, Convolutional Neural Network (CNN) [11] which is a type of a deep neural network is used for image identification and categorization. These are the algorithms that can recognize street signs, automobiles, people, and various other items. The real benefit of CNN is that it automatically detects the critical features after the training phase without any human intervention. Many CNN designs have been created to provide the most remarkable accuracy with increased processing speed.

The most popular and widely used CNN techniques are R-CNN (Region-based Convolutional Neural Networks) [12], Fast-RCNN [13], and Faster-RCNN [14]. However, the computational load was still too large for processing images on devices with limited computation, power, and space [15]. Therefore, the You Only Look Once (YOLO) model was developed to further improve the computation speed in classifying an object and determining its location in the image. It is based on a convolutional network framework to directly detect multiple objects within the image. It combines predictions from numerous feature maps with different resolutions to handle objects of various sizes [16]. YOLO kept on providing better performance in terms of processing time and accuracy due to the development of various new algorithms such as YOLOv3 and YOLOv5. The application of YOLO in the autonomous vehicle industry for object detection, localization, and classification in images and videos is presented in [17].

Object detection in a normal or autonomous environment may be affected by bad weather conditions such as hue or if it is too snowy or rainy [18]. In such cases, clear object recognition is complex, and therefore leads to the wrong judgment of vehicles or other objects on the road. In this case, various prediction-based previously trained models and algorithms are used to provide the proper assessment. To address the above challenges, the presented work intends to utilize the deep learning-based YOLOv5 algorithm for detecting and classifying vehicles on video from surveillance cameras and further processed using a deep learning algorithm in two different scenarios (with rain and without rain). We selected YOLOv5 since it is a well-known object detector that provides fast processing (improved computation speed) and is easy to train [19].

The localized road asset datasets were collected from different routes in Laval, Quebec City, Canada using four surveillance cameras installed on the vehicle's windshield for the required comprehensive analysis. The newly collected datasets were labeled for 11 different classes ('biker', 'car', 'pedestrian', 'trafficLight', 'trafficLight-Green', 'trafficLight-GreenLeft', 'trafficLight-Red', 'trafficLight-RedLeft', 'trafficLight-Yellow', 'trafficLight-YellowLeft', 'truck'). After that, the study involves training and evaluating YOLOv5 based deep neural network model considering two scenarios based on different combinations of the test and train datasets for detecting and classifying road assets. Finally, the performance of the prepared model is evaluated for two different weather conditions (with rain and without rain).

Overall, the contributions of this work can be listed as follows: Discussion on the various deep learning-based object detection techniques for vehicles and other road assets is presented in Section 2. Section 3 offers materials and methods, which include the proposed scheme, data sets applied for training and testing throughout experimentation. A detailed investigation of results and system performance is presented in Section 4. Section 5 provides the conclusion of the work with possible future guidelines

2. Related Work

A review of recent work has shown that image and video detection of vehicles can be enhanced using various machine learning algorithms. Cognitive Vehicles (CV) differ from Smart Vehicles (SV). They do not rely solely on sensor data and instead rigidly follow the patterns and functions that have already been preprogrammed externally. As a result, a new Global Navigation Satellite Systems (GNSS) free approach for vehicle self-localization has been developed [20]. Promising results are achieved when the system location estimations are compared to the GPS-reported locations. Authors in [21] developed a human detection system for intelligent surveillance in smart cities and societies based on Gaussian YOLOv3 method. Results showed that training enhances the Gaussian YOLOv3 algorithm's ability to detect humans, with an overall detection accuracy of 94%.

In [22], the authors presented a real-time road traffic management approach based on an upgraded YOLOv3. Using publicly available datasets, a neural network was trained and implemented the proposed strategy to improve vehicle detection. The evaluation findings demonstrated that the suggested system performed satisfactorily compared to the previous way of monitoring vehicle traffic. In addition, the proposed method was less expensive and had fewer hardware needs.

In [23], the authors presented a case study of YOLOv5 implementation to detect heavy goods vehicles in the winter, when there is snow, and in polar night situations. Results stated that a trained algorithm could see the front cabin of a heavy goods vehicle with high confidence; however, detecting the rear appeared more difficult, especially when the car is placed far away from the camera.

In [24], the primary learning models for video-based object detection that can be applied with autonomous vehicles are overviewed and investigated. The authors implemented a machine learning solution—the support vector machine (SVM) algorithm—and two deep learning solutions—the YOLO and the Single-Shot Multibox Detector (SSD) methods—in an autonomous vehicle environment. The drawback of the proposed method was that SVM performs poorly in simulations, and its speed did not match real-time response. In contrast, the YOLO model and SSD achieve greater accuracy and have a significant ability to detect objects in real-time when fast driving judgments are required. CNN-based YOLO provided better processing time and highly precise performance over time. The application of YOLO in the autonomous vehicle industry for object detection, localization, and classification in images and videos is presented in [25–27]. Other object recognition approaches in the vehicular environment under different weather conditions and traffic monitoring in real-time scenarios are investigated in [28–34].

We summarize the literature survey based on learning-based object detectors in Table 1 with the proposed scheme and their implementation challenges.

Table 1. Literature survey based on learning-based object detectors.

References	Proposed Scheme	Techniques Implemented	Advantages	Implementation Challenges
[22]	Real-time road traffic management is done using an improved YOLOv3 model.	Its a convolution neural network-based approach for the traffic analysis system, available online datasets are used to train the proposed neural network model, real video sequences of road traffic are used to test the performance of the proposed system.	The trained neural network improves vehicle detection, lowers cost, and has modest hardware requirements. Large-scale construction or installation work is not required.	Neural network-based model often produces detections with false rates due to incorrect input ranges (false positives).
[23]	Transfer learning to YOLOv5-based approach is utilized.	The proposed solution detects heavy goods vehicles at rest areas during winter to allow real-time prediction of parking spot occupancy in snowy conditions in winter.	Snowy conditions and the polar night in winter typically pose some challenges for image recognition; thermal network cameras can be used to solve the above problem.	The model faces some restrictions when analyzing images from small-angle cameras to detect objects that occur in groups and have a high number of overlaps and cut-offs.Detecting certain characteristic features of images can improve the model.
[24]	YOLOv4 network model is used to monitor traffic flow.	YOLOv4 network model is modified to increase the convolution times after the feature layer.	More global and higher semantic level feature information. More accurate than the original YOLOv4 model.	Increases the network complexity. The average detection time of the proposed model is slower than the original model.
[25–27]	Vehicle search is performed by detecting registration plates.	Neural networks, a block-difference method, and optical recognition techniques are used to detect moving objects.	Simplest in terms ofrecognition algorithms because of the contrast of the background and the characters, and the limited number of characters.	This approach does not allow detectingvehicles in situations where there are no license plates (bicycles) or when they arelocated in nonstandard areas (such as cars with temporary numbers).
[28–32]	Background subtraction-based implementation combined with blob analysis, Kalman filter, Gaussian Mixture Model (GMM).	Background subtraction: Vehicle detection is to segments moving implemented by subtracting the dynamic component (moving objects) from the static background of the image.	It is efficient for computation time and storage, and it is the simplest and most popular.	Processing data in dense traffic conditions lead to vehicle fusion due to partial occlusion in the processed image data. As a result, the prediction of an incorrect bounding box may occur.
[33]	Offline YOLO-based training method for object detection. Support vector machine is used to calculate the Haar wavelet function.	The offline tracker uses the detector for object detection in still images, and then a tracker based on Kalman filter associates the objects among video frames.	Offline YOLO trackers show more stability and provide improved performance. Faster with Kalman filter than the other trackers.	YOLO is not qualified for online tracking, because in this case it is very slow during the training phase.

Table 1. *Cont.*

References	Proposed Scheme	Techniques Implemented	Advantages	Implementation Challenges
[34]	An approach based on multilayer neural networks is used, and the network is trained by a new algorithm: Minimization of Inter-class Interference (MCI).	The proposed algorithm creates a hidden space (i.e., feature space) where the patterns have a desirable statistical distribution.	Simplicity and robustness enable real-time applications possible.	The neural architecture, the linear output layer, is replaced by the Mahalanobis kernel to improve generalization, and disturbing images are used; therefore, this approach is time-consuming.

3. Materials and Methods

A detailed explanation of the datasets and different experimentation results are presented in this section. The overall testing results are described in subsections; first, the performance of the pretrained algorithm is discussed. Secondly, imagery annotation and model training procedures are described. Finally, testing and validation using simulated datasets are done, and the algorithm performance is evaluated using different quantitative measures.

3.1. Proposed Scheme

We employed the Python programming language, the OpenCV image processing package, and the Google Colab cloud service in the suggested architecture. Python was chosen as the development programming language. A video stream processing method for recognizing objects and a tracking algorithm make up the internal subsystem. The YOLO neural network model, proven to be one of the most versatile and well-known object detection models, is used to process the data.

The advanced version of the YOLOv5 algorithm was used, which sends each batch of training data through the data loader while also improving the data. Scaling, color space correction, and mosaic enhancement are three types of data improvements that the data loader can execute.

This model uses the SxS grid system to separate all input images. Object detection is the responsibility of each grid. The boundary boxes for the detected object are now predicted by those Grid cells. These five key attributes are defined for each box: x and y for coordinates, w and h for object width and height, and a confidence score for the likelihood that the box contains the object. Additionally, YOLOv5 is faster when compared to YOLOv3, termed as more accurate. Another reason for using YOLOv5 for object detection is its fast processing time compared to YOLOv3.

In this paper, we provide a case study that shows how YOLOv5 can be used to recognize items on streets and highways, as well as object detection using YOLOv5 from Street-level Videos on 11 distinct classes: pedestrians, vehicles (car, truck, bike), and traffic signals (red, green, yellow). The workings of YOLOv5 with training and validation datasets and a tailored YOLOv5 model for the abovementioned class are shown in Figure 1.

3.2. Imagery Annotation and Model Training

The presented model was trained within the Google Colab cloud platform, with a powerful GPU tool that requires no configuration. We used a Roboflow self-driving car data set [35] built on YOLOv5 and employed pretrained COCO weights. The dataset was downloaded to Colab using the Roboflow generated URL as a zip folder. The overall annotated dataset was then split into a training set with 959 images, a validation set with 239 images, and a testing set with 302 images. Each image of Roboblow data was tagged with different classes. In this study, we trained our model for 11 different annotated classes (‘biker’, ‘car’, ‘pedestrian’, ‘trafficLight’, ‘trafficLight-Green’, ‘trafficLight-GreenLeft’, ‘trafficLight-Red’, 'trafficLight-RedLeft', 'trafficLight-Yellow', 'trafficLight-YellowLeft', 'truck'). It takes about 60 min to train the model.

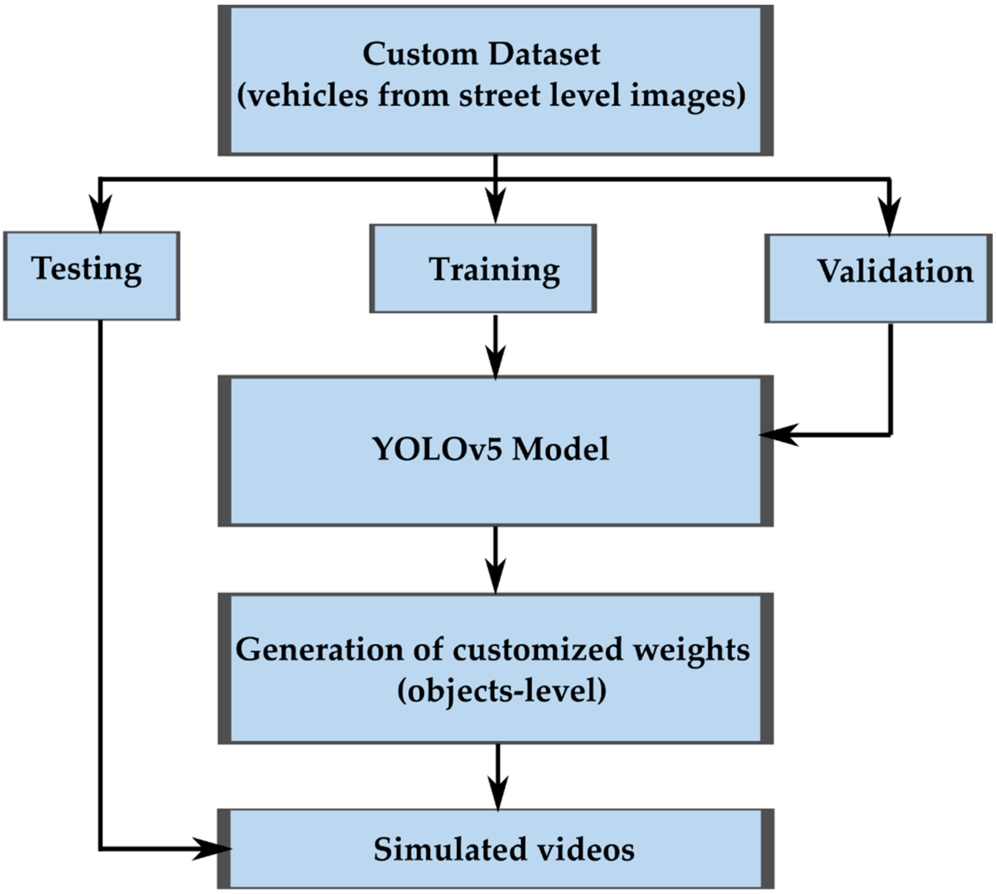

Figure 1. Object detection workflow using YOLOv5 model.

3.3. Testing and Validation Using Simulated Datasets

We tested and validated our model for two different scenarios: with rain and without rain (Figure 2). We prepared a simulated video of rain (Video S1) and without rain (Video S2) scenarios (Added videos in supplementary data). Then, we trained the YOLOv5 model using abovementioned Roboflow custom images with the help of custom data for 100 epochs. It took 18 min and 12 sec to complete 100 epochs. In the last step, both the simulated videos were validated using the best weights recorded during training of the YOLOv5 model. The main advantage of the YOLOv5 architecture is that objects are localized and classified in a single pass through the network. This allows for very quick frame-by-frame processing, making it possible to process video in real-time [35]. For detecting objects, three metrics named precision, recall, and mean AP (mAP) were used. Precision is calculated as the number of correctly marked objects divided by the total number of marked objects (error of commission). In contrast, recall is the number of correctly marked objects divided by the total number of objects present (error of omission) [36].

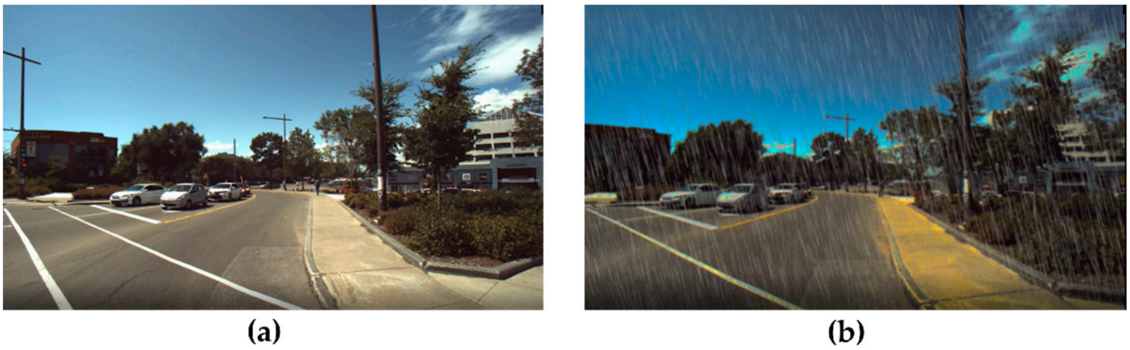

Figure 2. Extracted images from simulated video dataset showing (**a**) without rain and (**b**) with rain scenarios considered in this experiment.

4. Performance and Evaluation

It is clear from Figure 2a,b that the model can successfully detect all the specified classes with a high prediction value. The accuracy curves for precision and recall with confidence value and F1 score are plotted in Figure 3 (Supplementary data Videos S3 and S4).

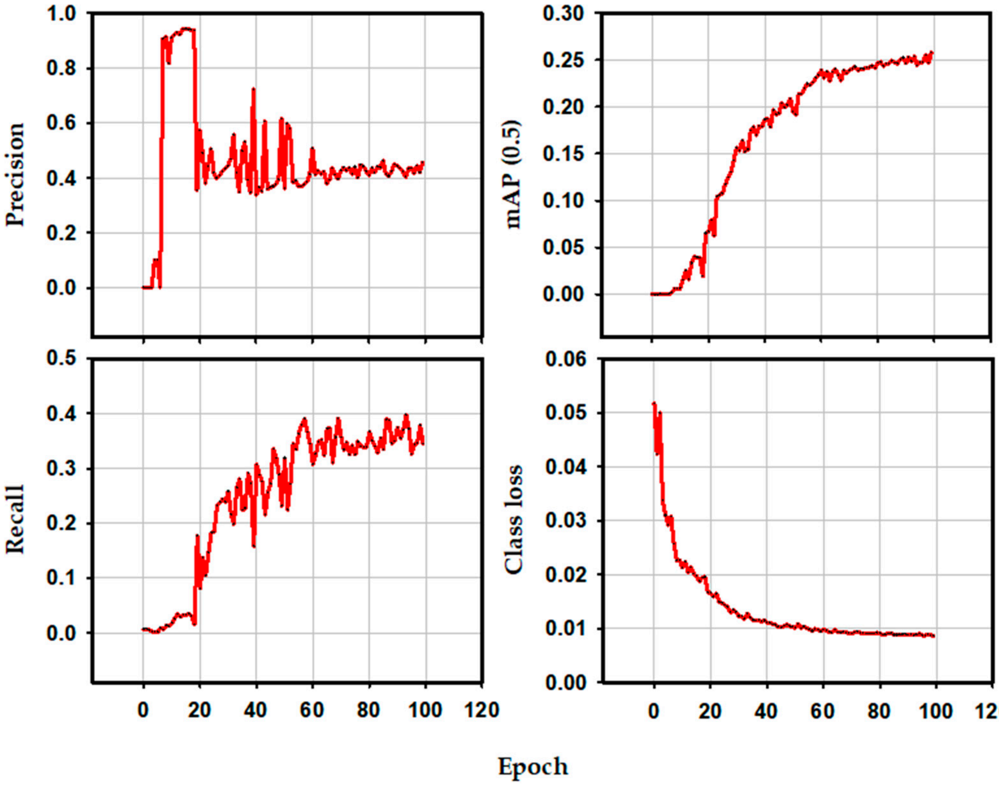

Figure 3. Plots of the precision, recall, mAP (0.5) parameters along with class object loss for training epochs.

The graphs in Figure 4 show the improvement in our model by displaying different performance metrics for both the training and validation sets. Figure 3 depicts classification loss. In this model, we used early stopping to select the best weights. The presented model shows improved precision, recall, and mAP until reaching a peak at 17, 93, and 99 epochs, respectively. The validation data's classification loss also showed a rapid decline after epoch 18. The loss function demonstrates how well a particular predictor performs in identifying the input data elements in a dataset. The lower the loss, the better the classifier models the relationship between input data and output targets. It displays how effectively the algorithm predicts the proper class of a given item in the situation of classification loss. Table 2 shows the results of those metrics for all classes, obtained on the first dataset with model YOLOv5s.

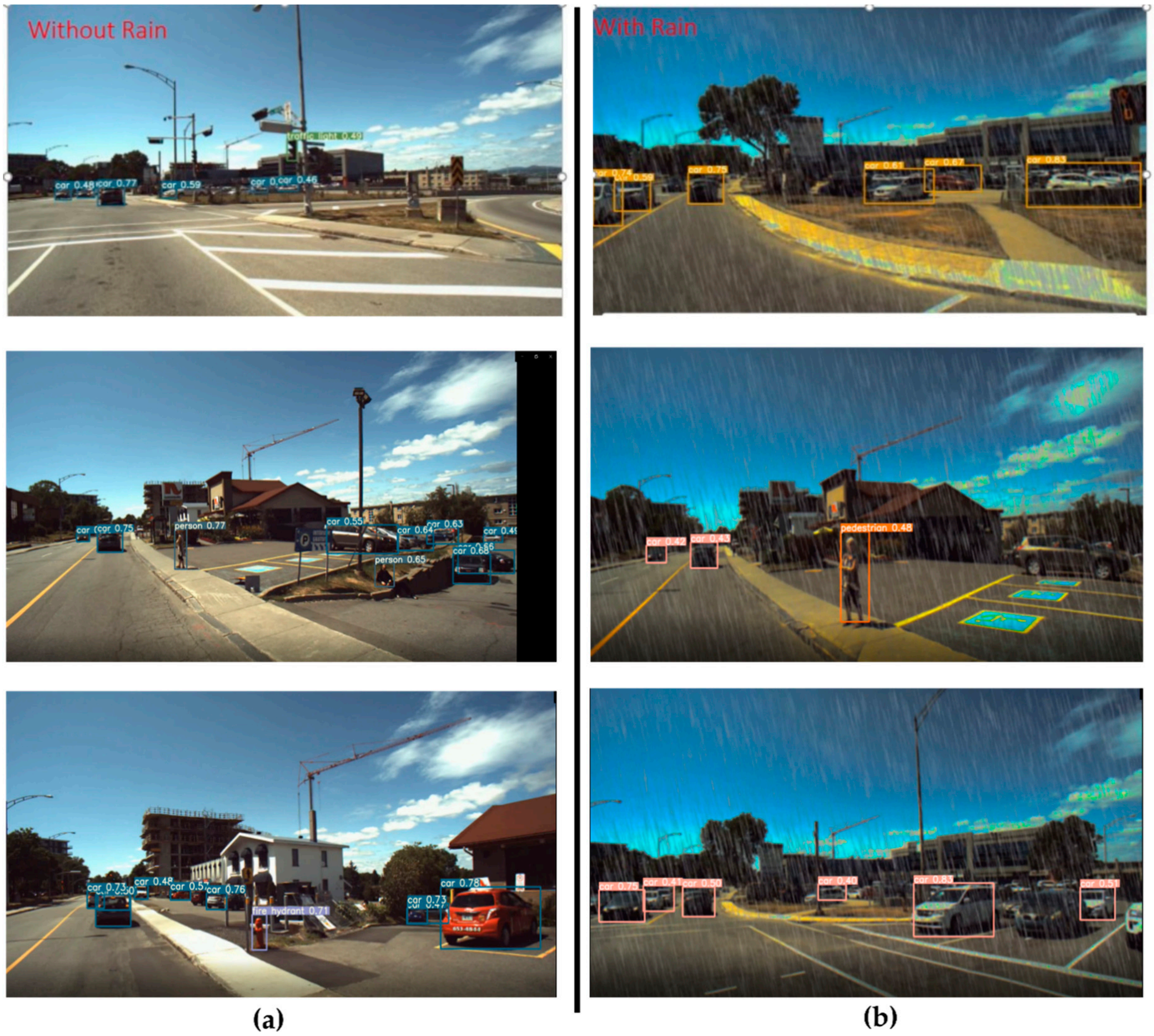

Figure 4. Model verification for two different scenarios: (**a**) without rain and (**b**) with rain.

Table 2 shows the results for each of the 11 classes and the entire validation set. The number of known targets to be detected is shown in the third column. The detector's accuracy and recall are shown in the fourth and fifth columns. Finally, the sixth column displays the mean average accuracy for the given intersection over the union. As the tables demonstrate, YOLOv5s performs similarly to the broader network. As a result, it is enough for the number of data and complexity of the problem, and larger models are not merited.

We see the most significant potential for improving performance in adjusting physical data collection and enhancing data annotation. For most applications, changes to the physical data collection cannot be influenced. However, as this is a pilot project running on only two different scenarios (rain and without rain), there is the possibility of changing the physical setup for data collection if more weather conditions are added.

This paper presents evidence that real-time camera videos captured while driving may be used as a test case for future studies. By incorporating a machine learning YOLOv5 model to detect real-time objects while driving on the road, we have essentially eliminated the bottleneck of image-by-image interpretation. We also showed that the proposed model performed better in precision and recall. Finally, our results showed that the presented approach can be used to investigate or identify different objects in developing and developed countries.

Table 2. Performance of the model YOLOv5s for custom Roboflow data.

Class	Images	Labels	Precision	Recall	mAP (0.5)
all	239	1520	0.474	0.337	0.258
biker	239	27	0.438	0.0741	0.0811
car	239	1066	0.528	0.726	0.723
pedestrian	239	156	0.22	0.308	0.186
trafficLight	239	41	0.308	0.415	0.297
trafficLight-Green	239	42	0.133	0.476	0.0956
trafficLight-GreenLeft	239	4	1	0	0.00756
trafficLight-Red	239	91	0.378	0.714	0.468
trafficLight-RedLeft	239	24	0.2	0.0833	0.128
trafficLight-Yellow	239	12	1	0	0.0169
truck	239	57	0.532	0.578	0.573

5. Conclusions

The tremendous expansion of urban infrastructure required a significant increase in the requirement for better road traffic management. In the literature, several strategies have been offered and discussed. This study provides a real-time road traffic management system based on an upgraded YOLOv5 model. We trained our model and implemented the proposed strategy to enhance vehicle recognition in rainy and regular weather conditions by utilizing an open dataset accessible at Roboflows. Rain and snow are challenging conditions for self-driving cars and often human drivers to deal with. Snow and rain impact the sensors and algorithms that control an autonomous vehicle.

In the same way that a skilled human driver can travel the same route in all weather, present autonomous cars are unable to generalize their experience in the same manner. We anticipate that self-driving cars will require more data to do this. The experimental findings showed that the YOLOv5 algorithm has an overall accuracy of 72.3% for car identification and 57.3% for truck identification for mAP (0.5). In the near future, this study can be applied in autonomous vehicle environments for various road assets detection in different weather conditions.

Supplementary Materials: The following supporting information can be downloaded at: https://www.mdpi.com/article/10.3390/electronics11040563/s1, Video S1: With rain scenario video, Video S2: Without rain scenario video, Video S3: Without rain scenario object detection video and Video S4: With rain scenario object detection video.

Author Contributions: Conceptualization, T.S., B.D., N.D., B.K. and A.C.; methodology, T.S.; software, T.S.; validation, T.S., B.D., N.D. and B.K.; formal analysis, T.S. and A.C.; investigation, T.S., B.D., N.D., B.K. and A.C.; resources, T.S., B.D., N.D. and B.K.; data curation, T.S., B.D., N.D. and B.K.; writing—original draft preparation, T.S.; writing—review and editing, T.S., A.C. and P.F. visualization, T.S., B.D., N.D. and B.K.; supervision, A.C. and P.F.; project administration, B.D. and N.D.; funding acquisition, B.D., N.D., A.C. and P.F. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by MITACS, grant number UBR 326853.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Due to the nature of this research, participants of this study did not agree for their data to be shared publicly, so supporting data is not available.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Bansal, P.; Kockelman, K.M. Are we ready to embrace connected and self-driving vehicles? A case study of Texans. *Transportation* **2018**, *45*, 641–675. [CrossRef]
2. Krishnaveni, P.; Sutha, J. Novel deep learning framework for broadcasting abnormal events obtained from surveillance applications. *J. Ambient Intell. Humaniz. Comput.* **2020**, 1–15. [CrossRef]
3. Ahad, M.A.; Paiva, S.; Tripathi, G.; Feroz, N. Enabling technologies and sustainable smart cities. *Sustain. Cities Soc.* **2020**, *61*, 102301. [CrossRef]
4. Chehri, H.; Chehri, A.; Saadane, R. Traffic signs detection and recognition system in snowy environment using deep learning. In *Proceedings of the Third International Conference on Smart City Applications*; Springer: Cham, Switzerland, 2020; pp. 503–513.
5. Peppa, M.V.; Bell, D.; Komar, T.; Xiao, W. Urban traffic flow analysis based on deep learning car detection from CCTV image series. In *Proceedings of the SPRS TC IV Mid-Term Symposium "3D Spatial Information Science–The Engine of Change"*; Newcastle University: Newcastle upon Tyne, UK, 2018.
6. Bahlmann, C.; Zhu, Y.; Ramesh, V.; Pellkofer, M.; Koehler, T. A system for traffic sign detection, tracking, and recognition using color, shape, and motion information. In Proceedings of the IEEE Proceedings Intelligent Vehicles Symposium, Las Vegas, NV, USA, 6–8 June 2005; pp. 255–260.
7. Pawełczyk, M.Ł.; Wojtyra, M. Real world object detection dataset for quadcopter unmanned aerial vehicle detection. *IEEE Access* **2020**, *8*, 174394–174409. [CrossRef]
8. Yahiaoui, M.; Rashed, H.; Mariotti, L.; Sistu, G.; Clancy, I.; Yahiaoui, L.; Kumar, V.R.; Yogamani, S. Fisheyemodnet: Moving object detection on surround-view cameras for autonomous driving. *arXiv* **2019**, arXiv:1908.11789.
9. Hu, L.; Ni, Q. IoT-driven automated object detection algorithm for urban surveillance systems in smart cities. *IEEE Internet Things J.* **2017**, *5*, 747–754. [CrossRef]
10. Bau, D.; Zhu, J.-Y.; Strobelt, H.; Lapedriza, A.; Zhou, B.; Torralba, A. Understanding the role of individual units in a deep neural network. *Proc. Natl. Acad. Sci. USA* **2020**, *117*, 30071–30078. [CrossRef] [PubMed]
11. Albawi, S.; Mohammed, T.A.; Al-Zawi, S. Understanding of a convolutional neural network. In Proceedings of the 2017 International Conference on Engineering and Technology (ICET), Antalya, Turkey, 21–23 August 2017; pp. 1–6.
12. Girshick, R.; Donahue, J.; Darrell, T.; Malik, J. YOLO9000: Better, Faster, Stronger. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, San Francisco, CA, USA, 18–20 June 1996; pp. 580–587.
13. Ren, S.; He, K.; Girshick, R.; Zhang, X.; Sun, J. Object Detection Networks on Convolutional Feature Maps. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 1476–1481. [CrossRef]
14. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 1137–1149. [CrossRef]
15. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, San Francisco, CA, USA, 18–20 June 1996; pp. 770–778.
16. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You only look once: Unified, real-time object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, San Francisco, CA, USA, 18–20 June 1996; pp. 779–788.
17. Gala, G.; Chavan, G.; Desai, N. Image Processing Based Driving Assistant System. *Iconic Res. Eng. J.* **2020**, *3*, 171–174.
18. Chehri, A.; Sharma, T.; Debaque, B.; Duclos, N.; Fortier, P. Transport Systems for Smarter Cities, a Practical Case Applied to Traffic Management in the City of Montreal. In *Sustainability in Energy and Buildings*; Springer: Singapore, 2021; pp. 255–266.
19. Delforouzi, A.; Pamarthi, B.; Grzegorzek, M. Training-based methods for comparison of object detection methods for visual object tracking. *Sensors* **2018**, *18*, 3994. [CrossRef] [PubMed]
20. Hayouni, A.; Debaque, B.; Duclos-Hindié, N.; Florea, M. Towards Cognitive Vehicles: GNSS-free Localization using Visual Anchors. In Proceedings of the 2020 IEEE 23rd International Conference on Information Fusion (FUSION), Rustenburg, South Africa, 6–9 July 2020; pp. 1–8.
21. Ahmed, I.; Jeon, G.; Chehri, A.; Hassan, M.M. Adapting Gaussian YOLOv3 with transfer learning for overhead view human detection in smart cities and societies. *Sustain. Cities Soc.* **2021**, *70*, 102908. [CrossRef]
22. Al-qaness, M.A.A.; Abbasi, A.A.; Fan, H.; Ibrahim, R.A.; Alsamhi, S.H.; Hawbani, A. An improved YOLO-based road traffic monitoring system. *Computing* **2021**, *103*, 211–230. [CrossRef]
23. Kasper-Eulaers, M.; Hahn, N.; Berger, S.; Sebulonsen, T.; Myrland, Ø.; Kummervold, P.E. Detecting Heavy Goods Vehicles in Rest Areas in Winter Conditions Using YOLOv5. *Algorithms* **2021**, *14*, 114. [CrossRef]
24. Yang, Y.; Cai, L.; Wei, H.; Qian, T.; Gao, Z. Research on Traffic Flow Detection Based on Yolo V4. In Proceedings of the 2021 16th International Conference on Computer Science & Education (ICCSE), Lancaster, UK, 17–21 August 2021; pp. 475–480.
25. Lee, H.-J.; Chen, S.-Y.; Wang, S.-Z. Extraction and recognition of license plates of motorcycles and vehicles on highways. In Proceedings of the Proceedings of the 17th International Conference on Pattern Recognition, Cambridge, UK, 26 August 2004; IEEE: Piscataway, NJ, USA, 2004; Volume 4, pp. 356–359.
26. De Oliveira, M.B.W.; Neto, A.D.A. Optimization of traffic lights timing based on multiple neural networks. In Proceedings of the 2013 IEEE 25th International Conference on Tools with Artificial Intelligence, Herndon, VA, USA, 4–6 November 2013; IEEE: Piscataway, NJ, USA, 2013; pp. 825–832.
27. Comelli, P.; Ferragina, P.; Granieri, M.N.; Stabile, F. Optical recognition of motor vehicle license plates. *IEEE Trans. Veh. Technol.* **1995**, *44*, 790–799. [CrossRef]

28. Dharamadhat, T.; Thanasoontornlerk, K.; Kanongchaiyos, P. Tracking object in video pictures based on background subtraction and image matching. In Proceedings of the 2008 IEEE International Conference on Robotics and Biomimetics, Washington, DC, USA, 22–25 February 2009; IEEE: Piscataway, NJ, USA, 2009; pp. 1255–1260.
29. Cancela, B.; Ortega, M.; Penedo, M.G.; Fernández, A. Solving multiple-target tracking using adaptive filters. In *Proceedings of the International Conference Image Analysis and Recognition*; Springer: Berlin/Heidelberg, Germany, 2011; pp. 416–425.
30. Sekar, G.; Deepika, M. Complex background subtraction using kalman filter. *Int. J. Eng. Res. Appl.* **2015**, *5*, 15–20.
31. Rabiu, H. Vehicle detection and classification for cluttered urban intersection. *Int. J. Comput. Sci. Eng. Appl.* **2013**, *3*, 37. [CrossRef]
32. Wang, K.; Liang, Y.; Xing, X.; Zhang, R. Target detection algorithm based on gaussian mixture background subtraction model. In *Proceedings of the 2015 Chinese Intelligent Automation Conference*; Springer: Berlin/Heidelberg, Germany, 2015; pp. 439–447.
33. Sun, Z.; Bebis, G.; Miller, R. Monocular precrash vehicle detection: Features and classifiers. *IEEE Trans. Image Process.* **2006**, *15*, 2019–2034. [PubMed]
34. Junior, O.L.; Nunes, U. Improving the generalization properties of neural networks: An application to vehicle detection. In Proceedings of the 2008 11th International IEEE Conference on Intelligent Transportation Systems, Beijing, China, 12–15 October 2008; IEEE: Piscataway, NJ, USA, 2008; pp. 310–315.
35. Roboflow How to Train YOLOv5 on Custom Objects. Available online: https://public.roboflow.com/object-detection/self-driving-car (accessed on 5 April 2020).
36. Fang, Y.; Guo, X.; Chen, K.; Zhou, Z.; Ye, Q. Accurate and Automated Detection of Surface Knots on Sawn Timbers Using YOLO-V5 Model. *BioResources* **2021**, *16*, 5390–5406. [CrossRef]